

JP00/01543

PCT/JP 00/01543

07.04.00 #5

日 本 国 特 許 庁

PATENT OFFICE
JAPANESE GOVERNMENT

④

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日
Date of Application: 1999年 3月15日

出 願 番 号
Application Number: 平成11年特許願第068407号

出 願 人
Applicant (s): 松下電器産業株式会社

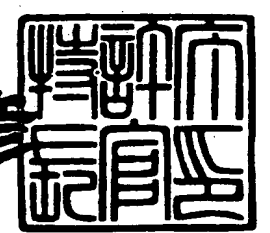
REC'D 26 MAY 2000
WIPO PCT

PRIORITY
DOCUMENT
SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH RULE 17.1(a) OR (b)

2000年 5月12日

特許庁長官
Commissioner,
Patent Office

近 藤 隆 彦



出証番号 出証特2000-3034901

【書類名】 特許願

【整理番号】 2037810014

【提出日】 平成11年 3月15日

【あて先】 特許庁長官殿

【国際特許分類】 H03M 13/22

【発明の名称】 ブロックインタリーブ装置, ブロックデインタリーブ装置, ブロックインタリーブ方法, およびブロックデインタリーブ方法

【請求項の数】 20

【発明者】

【住所又は居所】 大阪府門真市大字門真 1 0 0 6 番地 株式会社松下ソフトリサーチ内

【氏名】 古谷 専一

【特許出願人】

【識別番号】 000005821

【氏名又は名称】 松下電器産業株式会社

【代理人】

【識別番号】 100081813

【弁理士】

【氏名又は名称】 早瀬 憲一

【電話番号】 06(6380)5822

【手数料の表示】

【予納台帳番号】 013527

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9600402

【書類名】 明細書

【発明の名称】 ブロックインタリーブ装置、ブロックデインタリーブ装置、ブロックインタリーブ方法、およびブロックデインタリーブ方法

【特許請求の範囲】

【請求項1】 アドレスが連続する順序で割り当てられた記憶装置に対し1アドレス以上の同一間隔でアドレスを飛ばして書き込みを行い、

読み出しデータがブロックインタリーブされたものとなるような順序で読み出しを行って出力データとし、

その後この読み出しと同じ順序で書き込みを行うことによりブロックインタリーブを行うようにしたことを特徴とするブロックインタリーブ装置。

【請求項2】 請求項1記載のブロックインタリーブ装置において、本ブロックインタリーブ装置の出力を生成するための $L \times M$ データの記憶装置と、

前記記憶装置に出力するアドレスを生成するアドレス生成装置と、

前記記憶装置に対し制御信号を出力する記憶装置制御装置とを備え、

前記アドレス生成装置において、ブロック番号 b (b は0以上の整数)の0番目のアドレス $Ab(0)$ を0とし、ブロック番号 b の n 番目 (n は0以上の整数)のアドレス $Ab(n)$ を、 α (α は、2以上の整数)と M の $(b-x)$ 乗 (x は0以上 b 以下の整数)の乗算結果に $Ab(n-1)$ を加算した結果を $L \times M - 1$ で割った剰余から生成し、

この生成したアドレスに対して読み出し、書き込みを繰り返すことによりブロックインタリーブを行うことを特徴とするブロックインタリーブ装置。

【請求項3】 請求項2記載のブロックインタリーブ装置において、前記 α と $L \times M - 1$ との間に公約数が存在しないように、これらの値を設定したことを特徴とするブロックインタリーブ装置。

【請求項4】 請求項2記載のブロックインタリーブ装置において、前記 α と M の $(-x)$ 乗が等しくないように、これらの値を設定したことを特徴とするブロックインタリーブ装置。

【請求項5】 請求項2記載のブロックインタリーブ装置において、
前記 α 、 L 、 M の値を、それぞれ $\alpha=20$ 、 $L=8$ 、 $M=203$ としたことを特徴とするブロックインタリーブ装置。

【請求項6】 アドレスが連続する順序で割り当てられた記憶装置に対し1
アドレス以上の同一間隔でアドレスを飛ばして書き込みを行い、
読み出しデータがブロックデインタリーブされたものとなるような順序で読み
出しを行って出力データとし、

その後この読み出しと同じ順序で書き込みを行うことによりブロックデインタ
リーブを行うようにしたことを特徴とするブロックデインタリーブ装置。

【請求項7】 請求項6記載のブロックデインタリーブ装置において、
本ブロックデインタリーブ装置の出力を生成するための $L \times M$ データの記憶装
置と、

前記記憶装置に出力するアドレスを生成するアドレス生成装置と、

前記記憶装置に対し制御信号を出力する記憶装置制御装置とを備え、

前記アドレス生成装置において、ブロック番号 b (b は0以上の整数)の0番目
のアドレス $Ab(0)$ を0とし、ブロック番号 b の n 番目(n は0以上の整数)のア
ドレス $Ab(n)$ を α (α は、2以上の整数)と L の $(b-x)$ 乗(x は0以上 b 以下
の整数)の乗算結果に $Ab(n-1)$ を加算した結果を $L \times M-1$ で割った剰余から生成
し、

この生成したアドレスに対して読み出し、書き込みを繰り返すことによりプロ
ックデインタリーブを行うことを特徴とするブロックデインタリーブ装置。

【請求項8】 請求項7記載のブロックデインタリーブ装置において、
前記 α と $L \times M-1$ との間に公約数が存在しないように、これらの値を設定した
ことを特徴とするブロックデインタリーブ装置。

【請求項9】 請求項7記載のブロックデインタリーブ装置において、
前記 α と L の $(-x)$ 乗が等しくないように、これらの値を設定したことを特徴と
するブロックデインタリーブ装置。

【請求項10】 請求項7記載のブロックデインタリーブ装置において、
前記 α 、 L 、 M の値を、それぞれ $\alpha=20$ 、 $L=8$ 、 $M=203$ としたことを特徴とす

るブロックインタリーブ装置。

【請求項 11】 アドレスが連続する順序で割り当てられた記憶装置に対し
1 アドレス以上の同一間隔でアドレスを飛ばして書き込みを行い、

読み出しデータがブロックインタリーブされたものとなるような順序で読み出しを行って出力データとし、

その後この読み出しと同じ順序で書き込みを行うことによりブロックインタリーブを行うようにしたことを特徴とするブロックインタリーブ方法。

【請求項 12】 請求項 11 記載のブロックインタリーブ方法において、
ブロック番号 b (b は 0 以上の整数) の 0 番目のアドレス $Ab(0)$ を 0 とし、ブロック番号 b の n 番目 (n は 0 以上の整数) のアドレス $Ab(n)$ を α (α は、2 以上の整数) と M の $(b-x)$ 乗 (x は 0 以上 b 以下の整数) の乗算結果に $Ab(n-1)$ を加算した結果を $L \times M - 1$ で割った剰余から生成し、この生成したアドレスに対して $L \times M$ データの記憶装置の読み出し、書き込みを繰り返すことによりブロックインタリーブを行うことを特徴とするブロックインタリーブ方法。

【請求項 13】 請求項 12 記載のブロックインタリーブ方法において、
前記 α と $L \times M - 1$ との間に公約数が存在しないように、これらの値を設定したことを特徴とするブロックインタリーブ方法。

【請求項 14】 請求項 12 記載のブロックインタリーブ方法において、
前記 α と M の $(-x)$ 乗が等しくないように、これらの値を設定したことを特徴とするブロックインタリーブ方法。

【請求項 15】 請求項 12 記載のブロックインタリーブ方法において、
前記 α 、 L 、 M の値を、それぞれ $\alpha=20$ 、 $L=8$ 、 $M=203$ としたことを特徴とするブロックインタリーブ方法。

【請求項 16】 アドレスが連続する順序で割り当てられた記憶装置に対し
1 アドレス以上の同一間隔でアドレスを飛ばして書き込みを行い、

読み出しデータがブロックインタリーブされたものとなるような順序で読み出しを行って出力データとし、

その後この読み出しと同じ順序で書き込みを行うことによりブロックインタリーブを行うようにしたことを特徴とするブロックインタリーブ方法。

【請求項 17】 請求項 16 記載のブロックデインタリーブ方法において、ブロック番号 b (b は 0 以上の整数) の 0 番目のアドレス $Ab(0)$ を 0 とし、ブロック番号 b の n 番目 (n は 0 以上の整数) のアドレス $Ab(n)$ を α (α は、2 以上の整数) と L の $(b-x)$ 乗 (x は 0 以上 b 以下の整数) の乗算結果に $Ab(n-1)$ を加算した結果を $L \times M-1$ で割った剰余から生成し、生成したアドレスに対して $L \times M$ データの記憶装置の読み出し、書き込みを繰り返すことによりブロックデインタリーブを行うことを特徴とするブロックデインタリーブ方法。

【請求項 18】 請求項 17 記載のブロックデインタリーブ方法において、前記 α と $L \times M-1$ との間に公約数が存在しないように、これらの値を設定したことを特徴とするブロックデインタリーブ方法。

【請求項 19】 請求項 17 記載のブロックデインタリーブ方法において、前記 α と L の $(-x)$ 乗が等しくないように、これらの値を設定したことを特徴とするブロックデインタリーブ方法。

【請求項 20】 請求項 17 記載のブロックデインタリーブ方法において、前記 α 、 L 、 M の値を、それぞれ $\alpha=20$ 、 $L=8$ 、 $M=203$ としたことを特徴とするブロックデインタリーブ方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

この発明は、衛星放送・地上波放送・ケーブル放送などのデジタル伝送及びハードディスクなどの記憶装置の読み出し・書き込みにおいて必要とされるブロックインタリーブ装置、ブロックデインタリーブ装置、ブロックインタリーブ方法、およびブロックデインタリーブ方法に関するものである。

【0002】

【従来の技術】

ブロックインタリーブはバースト誤りに対する対策として有効なものである。

【0003】

これを、衛星放送を例にとって説明すると、地上の放送局からの電波は衛星に送出され、衛星で中継されて、家庭に設けた衛星放送受信機によって受信される

【0004】

ここで、放送局から衛星を介して家庭に向けて送信された電波は、伝送経路中で雷や雨等があると妨害を受け、その妨害を受けている間中、データに誤りが生じてしまう。これがバースト誤りと言われるものである。

【0005】

デジタル伝送では予めエラー訂正用の情報が本来のデータに付加されているので、各セグメントにおいて一定範囲内のビットの誤りであればこれを訂正することが可能であるが、バースト誤りのような連続した誤りはこれを訂正することができない。

【0006】

そこで、予め伝送すべきデータを時間的に分散させておくことにより（その手法がブロックインタリーブである）、伝送の過程でバースト誤りが生じたとしても、受信側で予め分散されていたデータの時間的位置を復元する際に（その手法がブロックデインタリーブである）、バースト誤りが分散され、個々のデータブロックでは誤りをエラー訂正可能な範囲のビット数に収まるようにすることができる。

【0007】

このようなブロックインタリーブやブロックデインタリーブを行おうとすれば、本来1ブロック(L×M データ)の記憶領域を持つ記憶装置2面を用いてこれらの記憶装置に書き込みと読み出しを交互に繰り返す処理が必要になるが、これを小回路規模かつ低消費電力で実現できるものの1つとして、従来、例えば特表平8-511393号公報に示されたものがあった。

【0008】

図13はこの従来のブロックインタリーブの動作を模式的に示すものであり、ここでは、説明の簡単化のために、4行5列のデータに対し、ブロックインタリーブを行う場合を例にとって説明する。

この方式は、ブロックインタリーブ装置の記憶装置のアドレスが図13(a)のように割り当てられているとすると、まず、REGを1とし、 $0 \rightarrow 1 \rightarrow 2 \rightarrow \dots$

→19の順で、すなわちそのアドレスが増加する順に従ってデータを書き込んでゆく。次に、図13(b)に示されているように、アドレスが増加する順に従ってデータを読み出す。即ち、REGに5を乗じ、図13(a)におけるアドレス0を初期値として、1入力データ毎に5増加するアドレスを順次生成するが、その際、アドレスが19($=4 \times 5 - 1$)を越えるところの19を減算する。そして、この生成規則に従って生成したアドレスに従い、まず、図13(a)において既に書き込まれているデータを、この生成したアドレスの順に読み出し、その読み出しが終了した後に、読み出しを行ったのと同じ順でデータの書き込みを行なう。

【0009】

次に、図13(c)に示されているように、このREGに対し5を乗じるが、その値が19を越えるところの19を減算する。

そして、図13(a)に示されているアドレスの配置に対しアドレス0を初期値として、1入力データ毎に6増加するアドレスを順次生成し、アドレスが19($=4 \times 5 - 1$)を越えるところ、この19を減算することにより読み出しを行い、その読み出しが終了した後に、読み出しを行ったのと同じ順でデータの書き込みを行なう。

【0010】

以下、同様の処理を繰り返すことにより、順次異なるアドレスの順で読み出し、書き込みを行ってゆけば、この例では図13(j)の時点において、図13(a)と同様のアドレスの順に戻る。

このような手順を繰り返すことにより、図14に示すように、1ブロック(L×M データ)の記憶領域を持つRAM202を用いてブロックインタリーブを行うことが可能である。これはRAM制御装置200による書き込み読み出し制御、およびアドレス生成装置201により生成するアドレスを上述のように工夫することにより実現しているものである。

【0011】

この従来例のブロックインタリーブ装置におけるアドレス生成規則は以下の通りである。

即ち、n番目のアドレスを $Ab(n)$ 、記憶装置の行の数をL、列の数をM、b

を0以上の整数、 x を0以上 b 以下の任意の整数とすると、

$$Ab(n) = (Ab(n-1) + M * (b-x)) \bmod (L \times M - 1) \dots (1)$$

$$\text{また、} REG = (M * (b-x)) \bmod (L \times M - 1)$$

ただし、 $Ab(0) = 0$ とする。また、 $*$ はべき乗を表わす。

【0012】

また、このようにしてブロックインタリーブを行ったデータに対しブロックデインタリーブを行うには、

ブロックデインタリーブ装置の記憶装置のアドレスが図13(k)のように割り当てられているとすると、まず、 REG を1とし、 $0 \rightarrow 1 \rightarrow 2 \rightarrow \dots \rightarrow 19$ の順で、すなわちそのアドレスが増加する順に従ってデータを書き込んでゆく。次に、図13(l)に示されているように、データを読み出す。即ち、 REG に4を乗じ、図13(k)におけるアドレス0を初期値として、1入力データ毎に4増加するアドレスを順次生成するが、その際、アドレスが19($=4 \times 5 - 1$)を越えるとこの19を減算する。そして、この生成規則に従って生成したアドレスに従い、まず、図13(k)において既書き込まれているデータを、この生成したアドレスの順に読み出し、その読み出しが終了した後に、読み出しを行ったのと同じ順でデータの書き込みを行なう。

【0013】

次に、図13(m)に示されているように、この REG に対し4を乗じるが、その値が19を越えるとこの19を減算する。

そして、図13(k)に示されているアドレスの配置に対しアドレス0を初期値として、1入力データ毎に16増加するアドレスを順次生成し、アドレスが19($=4 \times 5 - 1$)を越えるとこの19を減算することにより読み出しを行い、その読み出しが終了した後に、読み出しを行ったのと同じ順でデータの書き込みを行なう。

【0014】

以下、同様の処理を繰り返すことにより、順次異なるアドレスの順で読み出し、書き込みを行ってゆけば、この例では図13(t)の時点において、図13(k)と同様のアドレスの順に戻る。

このような手順を繰り返すことにより、図14に示すように、1ブロックの記憶領域を持つRAM202を用いてブロックデインタリーブを行うことが可能である。これはRAM制御装置200による書き込み読み出し制御、およびアドレス生成装置201により生成するアドレスを上述のように工夫することにより実現しているものである。

【0015】

この従来例のブロックデインタリーブ装置におけるアドレス生成規則は以下の通りである。

$$Ab(n) = (Ab(n-1) + L * (b-x)) \bmod (L \times M - 1) \dots (2)$$

$$\text{また、} REG = (L * (b-x)) \bmod (L \times M - 1)$$

ただし、 $Ab(0) = 0$ とする。

この式(2)は式(1)のMをLに変更したものである。

【0016】

【発明が解決しようとする課題】

従来のブロックインタリーブ装置およびブロックデインタリーブ装置は以上のように構成されており、上述のように1ブロックの記憶領域を持つ記憶装置1面のみを用いてブロックインタリーブおよびブロックデインタリーブを行うことができ、装置の省面積化、および低消費電力化を実現できるものである。

しかしながら、本件発明者が鋭意研究を行った結果、この従来の処理方式には、さらに回路面積を小面積化でき、あるいは消費電力を低減できる余地があることが判明した。

【0017】

本発明は、記憶装置制御装置を最適化することにより、回路面積のより省面積化や、より低消費電力化を実現できるブロックインタリーブ装置、ブロックデインタリーブ装置、ブロックインタリーブ方法、およびブロックデインタリーブ方法を得ることを目的とする。

【0018】

【課題を解決するための手段】

本願の請求項1の発明に係るブロックインタリーブ装置は、アドレスが連続す

る順序で割り当てられた記憶装置に対し1アドレス以上の同一間隔でアドレスを飛ばして書き込みを行い、読み出しデータがブロックインタリーブされたものとなるような順序で読み出しを行って出力データとし、その後この読み出しと同じ順序で書き込みを行うことによりブロックインタリーブを行うようにしたものである。

本願の請求項1の発明に係るブロックインタリーブ装置は、上述のような書き込み、読み出し動作を行うことにより、1ブロックの記憶領域を持つ記憶装置1面でのブロックインタリーブ動作を可能とし、かつ記憶装置制御装置の回路規模を縮小可能とする。

【0019】

また、本願の請求項2の発明に係るブロックインタリーブ装置は、請求項1記載のブロックインタリーブ装置において、本ブロックインタリーブ装置の出力を生成するための $L \times M$ データの記憶装置と、前記記憶装置に出力するアドレスを生成するアドレス生成装置と、前記記憶装置に対し制御信号を出力する記憶装置制御装置とを備え、前記アドレス生成装置において、ブロック番号 b の0番目のアドレス $Ab(0)$ を0とし、ブロック番号 b の n 番目(n は0以上の整数)のアドレス $Ab(n)$ を、 α (α は、2以上の整数)と M の $(b-x)$ 乗(x は0以上 b 以下の整数)の乗算結果に $Ab(n-1)$ を加算した結果を $L \times M - 1$ で割った剰余から生成し、この生成したアドレスに対して読み出し、書き込みを繰り返すことによりブロックインタリーブを行うようにしたものである。

本願の請求項2の発明に係るブロックインタリーブ装置は、上述のように構成したことにより、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックインタリーブを行うことを可能とする。

【0020】

また、本願の請求項3の発明に係るブロックインタリーブ装置は、請求項2記載のブロックインタリーブ装置において、前記 α と $L \times M - 1$ との間に公約数が存在しないように、これらの値を設定したものである。

本願の請求項3の発明に係るブロックインタリーブ装置は、上述のように構成したことにより、アドレスの生成規則が破綻するのを防止し、記憶装置、アドレ

ス生成装置を最適化し、最小限の回路面積により、ブロックインタリーブを行うことを可能とする。

【0021】

また、本願の請求項4の発明に係るブロックインタリーブ装置は、請求項2記載のブロックインタリーブ装置において、前記 α とMの(-x)乗が等しくないように、これらの値を設定したものである。

本願の請求項4の発明に係るブロックインタリーブ装置は、上述のように構成したことにより、最初の書き込みの時点においてアドレスを連続して書き込みを行うのを防止し、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックインタリーブを行うことを可能とする。

【0022】

また、本願の請求項5の発明に係るブロックインタリーブ装置は、請求項2記載のブロックインタリーブ装置において、前記 α 、L、Mの値を、それぞれ $\alpha=20$ 、 $L=8$ 、 $M=203$ としたものである。

本願の請求項5の発明に係るブロックインタリーブ装置は、上述のように構成したことにより、アドレス生成装置を構成する比較器の面積が低減し、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックインタリーブを行うことを可能とする。

【0023】

また、本願の請求項6の発明に係るブロックデインタリーブ装置は、アドレスが連続する順序で割り当てられた記憶装置に対し1アドレス以上の同一間隔でアドレスを飛ばして書き込みを行い、読み出しデータがブロックデインタリーブされたものとなるような順序で読み出しを行って出力データとし、その後この読み出しと同じ順序で書き込みを行うことによりブロックデインタリーブを行うようにしたものである。

本願の請求項6の発明に係るブロックデインタリーブ装置は、上述のような書き込み、読み出し動作を行うことにより、1ブロックの記憶領域を持つ記憶装置1面でのブロックデインタリーブ動作を可能とし、かつ記憶装置制御装置の回路規模を縮小可能とする。

【0024】

また、本願の請求項7の発明に係るブロックデインタリーブ装置は、請求項6記載のブロックデインタリーブ装置において、本ブロックデインタリーブ装置の出力を生成するための $L \times M$ データの記憶装置と、前記記憶装置に出力するアドレスを生成するアドレス生成装置と、前記記憶装置に対し制御信号を出力する記憶装置制御装置とを備え、前記アドレス生成装置において、ブロック番号 b の0番目のアドレス $Ab(0)$ を0とし、ブロック番号 b の n 番目(n は0以上の整数)のアドレス $Ab(n)$ を、 α (α は、2以上の整数)と L の $(b-x)$ 乗(x は0以上 b 以下の整数)の乗算結果に $Ab(n-1)$ を加算した結果を $L \times M-1$ で割った剰余から生成し、この生成したアドレスに対して読み出し、書き込みを繰り返すことによりブロックデインタリーブを行うようにしたものである。

本願の請求項7の発明に係るブロックデインタリーブ装置は、上述のように構成したことにより、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックデインタリーブを行うことを可能とする。

【0025】

また、本願の請求項8の発明に係るブロックデインタリーブ装置は、請求項7記載のブロックデインタリーブ装置において、前記 α と $L \times M-1$ との間に公約数が存在しないように、これらの値を設定したものである。

本願の請求項8の発明に係るブロックデインタリーブ装置は、上述のように構成したことにより、アドレスの生成規則が破綻するのを防止し、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックデインタリーブを行うことを可能とする。

【0026】

また、本願の請求項9の発明に係るブロックデインタリーブ装置は、請求項7記載のブロックデインタリーブ装置において、前記 α と L の $(-x)$ 乗が等しくないように、これらの値を設定したものである。

本願の請求項9の発明に係るブロックデインタリーブ装置は、上述のように構成したことにより、最初の書き込みの時点においてアドレスを連続して書き込みを行うのを防止し、記憶装置、アドレス生成装置を最適化し、最小限の回路面積

により、ブロックデインタリーブを行うことを可能とする。

【 0 0 2 7 】

また、本願の請求項 1 0 の発明に係るブロックデインタリーブ装置は、請求項 7 記載のブロックデインタリーブ装置において、前記 α 、 L 、 M の値を、それぞれ $\alpha=20$ 、 $L=8$ 、 $M=203$ としたものである。

本願の請求項 1 0 の発明に係るブロックデインタリーブ装置は、上述のように構成したことにより、アドレス生成装置を構成する比較器の面積が低減し、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックデインタリーブを行うことを可能とする。

【 0 0 2 8 】

また、本願の請求項 1 1 の発明に係るブロックインタリーブ方法は、アドレスが連続する順序で割り当てられた記憶装置に対し 1 アドレス以上の同一間隔でアドレスを飛ばして書き込みを行い、読み出しデータがブロックインタリーブされたものとなるような順序で読み出しを行って出力データとし、その後この読み出しと同じ順序で書き込みを行うことによりブロックインタリーブを行うようにしたものである。

本願の請求項 1 1 の発明に係るブロックインタリーブ方法は、上述のような書き込み、読み出し動作を行うことにより、1 ブロックの記憶領域を持つ記憶装置 1 面でのブロックインタリーブ動作を可能とし、かつ記憶装置制御装置の回路規模を縮小可能とする。

【 0 0 2 9 】

また、本願の請求項 1 2 の発明に係るブロックインタリーブ方法は、請求項 1 1 記載のブロックインタリーブ方法において、ブロック番号 b の 0 番目のアドレス $Ab(0)$ を 0 とし、ブロック番号 b の n 番目 (n は 0 以上の整数) のアドレス $Ab(n)$ を、 α (α は、2 以上の整数) と M の $(b-x)$ 乗 (x は 0 以上 b 以下の整数) の乗算結果に $Ab(n-1)$ を加算した結果を $L \times M - 1$ で割った剰余から生成し、この生成したアドレスに対して $L \times M$ データの記憶装置の読み出し、書き込みを繰り返すことによりブロックインタリーブを行うようにしたものである。

本願の請求項 1 2 の発明に係るブロックインタリーブ方法は、上述のように構

成したことにより、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックインタリーブを行うことを可能とする。

【0030】

また、本願の請求項13の発明に係るブロックインタリーブ方法は、請求項12記載のブロックインタリーブ方法において、前記 α と $L \times M - 1$ との間に公約数が存在しないように、これらの値を設定するようにしたものである。

本願の請求項13の発明に係るブロックインタリーブ方法は、上述のように構成したことにより、アドレスの生成規則が破綻するのを防止し、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックインタリーブを行うことを可能とする。

【0031】

また、本願の請求項14の発明に係るブロックインタリーブ方法は、請求項12記載のブロックインタリーブ方法において、前記 α と M の $(-x)$ 乗が等しくないように、これらの値を設定するようにしたものである。

本願の請求項14の発明に係るブロックインタリーブ方法は、上述のように構成したことにより、最初の書き込みの時点においてアドレスを連続して書き込みを行うのを防止し、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックインタリーブを行うことを可能とする。

【0032】

また、本願の請求項15の発明に係るブロックインタリーブ方法は、請求項12記載のブロックインタリーブ方法において、前記 α 、 L 、 M の値を、それぞれ $\alpha = 20$ 、 $L = 8$ 、 $M = 203$ としたものである。

本願の請求項15の発明に係るブロックインタリーブ方法は、上述のように構成したことにより、アドレス生成装置を構成する比較器の面積が低減し、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックインタリーブを行うことを可能とする。

【0033】

また、本願の請求項16の発明に係るブロックインタリーブ方法は、アドレスが連続する順序で割り当てられた記憶装置に対し1アドレス以上の同一間隔で

アドレスを飛ばして書き込みを行い、読み出しデータがブロックデインタリーブされたものとなるような順序で読み出しを行って出力データとし、その後この読み出しと同じ順序で書き込みを行うことによりブロックデインタリーブを行うようにしたものである。

本願の請求項 1 6 の発明に係るブロックデインタリーブ方法は、上述のような書き込み、読み出し動作を行うことにより、1 ブロックの記憶領域を持つ記憶装置 1 面でのブロックデインタリーブ動作を可能とし、かつ記憶装置制御装置の回路規模を縮小可能とする。

【0034】

また、本願の請求項 1 7 の発明に係るブロックデインタリーブ方法は、請求項 1 6 記載のブロックデインタリーブ方法において、ブロック番号 b の 0 番目のアドレス $Ab(0)$ を 0 とし、ブロック番号 b の n 番目 (n は 0 以上の整数) のアドレス $Ab(n)$ を、 α (α は、2 以上の整数) と L の $(b-x)$ 乗 (x は 0 以上 b 以下の整数) の乗算結果に $Ab(n-1)$ を加算した結果を $L \times M - 1$ で割った剰余から生成し、この生成したアドレスに対して $L \times M$ データの記憶装置の読み出し、書き込みを繰り返すことによりブロックデインタリーブを行うようにしたものである。

本願の請求項 1 7 の発明に係るブロックデインタリーブ方法は、上述のように構成したことにより、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックデインタリーブを行うことを可能とする。

【0035】

また、本願の請求項 1 8 の発明に係るブロックデインタリーブ方法は、請求項 1 7 記載のブロックデインタリーブ方法において、前記 α と $L \times M - 1$ との間に公約数が存在しないように、これらの値を設定したものである。

本願の請求項 1 8 の発明に係るブロックデインタリーブ方法は、上述のように構成したことにより、アドレスの生成規則が破綻するのを防止し、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックデインタリーブを行うことを可能とする。

【0036】

また、本願の請求項 1 9 の発明に係るブロックデインタリーブ方法は、請求項

17記載のブロックデインタリーブ方法において、前記 α と L の $(-x)$ 乗が等しくないように、これらの値を設定するようにしたものである。

本願の請求項19の発明に係るブロックデインタリーブ方法は、上述のように構成したことにより、最初の書き込みの時点においてアドレスを連続して書き込みを行うのを防止し、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックデインタリーブを行うことを可能とする。

【0037】

また、本願の請求項20の発明に係るブロックデインタリーブ方法は、請求項17記載のブロックデインタリーブ方法において、前記 α 、 L 、 M の値を、それぞれ $\alpha=20$ 、 $L=8$ 、 $M=203$ としたものである。

本願の請求項20の発明に係るブロックデインタリーブ方法は、上述のように構成したことにより、アドレス生成装置を構成する比較器の面積が低減し、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックデインタリーブを行うことを可能とする。

【0038】

【発明の実施の形態】

（実施の形態1）

以下、本発明の実施の形態1について、図を用いて説明する。

本願発明によるブロックインタリーブ装置およびブロックインタリーブ方法について説明する。

本実施の形態1のブロックインタリーブ装置およびブロックインタリーブ方法は、記憶装置のアドレス生成装置を最適化することにより、記憶装置の制御装置の面積あるいは消費電力をより低減できるようにしたものである。

【0039】

図1は本願発明の実施の形態1による、 $L \times M$ データのブロックインタリーブを行うブロックインタリーブ装置を示すものである。図1において、101は本ブロックインタリーブ装置によりブロックインタリーブを行うべき入力データの入力端子、114はこのブロックインタリーブを行うべき入力データのブロックの先頭入力データ毎に同期して入力される先頭入力データ同期信号の入力端子、

106は各入力データ毎に発生する同期信号の入力端子、112は同期信号入力端子106から入力される同期信号に応じて記憶装置104を制御する制御装置、103は入力端子106から入力される同期信号および入力信号114から入力される先頭入力データ同期信号に基づいて記憶装置104のアドレスを生成するアドレス生成装置、120はこのアドレス生成装置103が生成したアドレスを出力する出力端子、104は記憶装置であり、制御装置112の制御により、アドレス生成装置103により生成されたアドレスに入力端子101からの入力データを書き込み、これを読み出すことによりブロックインタリーブを行うものである。105は本ブロックインタリーブ装置によりインタリーブされたデータを出力する出力端子である。

【0040】

また、図1のアドレス生成装置103において、110は初期値Mを発生する定数発生器、111はこの初期値Mとレジスタ113の出力を乗算する乗算器、140はこの乗算器111の出力がオーバーフローした場合の処理を行うオーバーフロー処理装置、113はこのオーバーフロー処理装置140の出力または定数発生器118の出力がセットされるレジスタ、118は初期値 α を発生する定数発生器、115はこのレジスタ113の出力とレジスタ117の出力を加算する加算器、141はこの加算器115の出力がオーバーフローした場合の処理を行うオーバーフロー処理装置、117はこのオーバーフロー処理装置141の出力または定数発生器119の出力がセットされるレジスタ、119は初期値1を発生する定数発生器である。

図2は本実施の形態1によるブロックインタリーブ装置の動作を模式的に示すものであり、4行5列のデータに対し、ブロックインタリーブを行う場合を例にとって示している。

【0041】

次に図1の動作について、図2に示した、4行5列のデータに対し、ブロックインタリーブを行う場合を例にとって説明する。

図1に示すように、本実施の形態1によるブロックインタリーブ装置は、入力端子101から入力されるデータをL×Mデータの記憶装置104に書き込み、

このL×M データの記憶装置104からデータを読み出すことによりブロックインタリーブを行うものであり、その際、図2に示すような順序で書き込み、読み出しを行うように、制御装置112が前記記憶装置104へ制御信号を出力することによりその書き込み、読み出し制御を行い、かつアドレス生成装置103がその書き込み、読み出しの際のアドレスを生成し、これを前記記憶装置104に出力することにより、1ブロックの記憶領域を持つ記憶装置1面でブロックインタリーブされた出力105を生成できるものである。

【0042】

この装置は、ブロックインタリーブ装置の記憶装置104のアドレスが図13(a)のように割り当てられているとすると、まず、図2(a)のようにREGを2とし、図13(a)のアドレス0を初期値とし入力データ毎に2増加する書き込みアドレスを順次生成する。その際、書き込みアドレスが19($=4 \times 5 - 1$)を越えるとこの19を減算する。そして、この生成規則に従って生成した書き込みアドレスに従い、データを書き込みを、ブロック内の全てのアドレスにアクセスが完了する迄行なう。

即ち、図13(a)の従来方式では書き込みアドレスが増加する順に従って0→1→2→……→19の順でデータを順次書き込んでいたものが、本実施の形態1ではこれを1つ飛ばしに書き込んでゆくものである。

【0043】

次に、図2(b)に示されているように、REGに5を乗じ、図2(a)におけるアドレスの並びを基準として、図2(a)のアドレス0を初期値とし、入力データ毎にアドレスが10($=2 \times 5$)増加するアドレスを順次生成するが、その際、アドレスが19($=4 \times 5 - 1$)を越えるとこの19を減算する。

そして、この生成規則に従って生成したアドレスに従い読み出しおよび書き込みを、ブロック内の全てのアドレスにアクセスが完了する迄行なう。

【0044】

次に、図2(c)に示されているように、このREGに対し5を乗じるが、その値が19を越えているのでこの19により除算を行いその剰余12を求める。

そして、図13(a)に示されているアドレスの並びを基準とし、そのアドレス

0を初期値として、入力データ毎に12増加するアドレスを順次生成し、アドレスが19(=4×5-1)を越えるとこの19を減算する。

そして、この生成規則に従って生成したアドレスに従い読み出しおよび書き込みを、ブロック内の全てのアドレスにアクセスが完了する迄行なう。

【0045】

以下、同様の処理を繰り返すことにより、順次異なるアドレスの順で読み出し、書き込みを行ってゆけば、この例では図2(j)の時点において、図2(a)と同様のアドレスの順に戻る。

このような手順を繰り返すことにより、図3に示すように、1ブロックの記憶領域を持つ記憶装置1面を用いてブロックインタリーブを行うことを可能としている。これは制御装置112による書き込み読み出し制御、およびアドレス生成装置103により生成する記憶装置104のアドレスを上述のように工夫することにより実現しているものであり、これに加えて、本実施の形態1ではアドレス生成装置の回路面積や消費電力の低減が可能になっているものである。

【0046】

この実施の形態1におけるアドレス生成規則は以下の通りである。

即ち、n番目のアドレスを $Ab(n)$ 、記憶装置の行の数をL、列の数をM、ブロック番号bを0以上の整数、xを0以上b以下の任意の整数とすると、

$$Ab(n) = (Ab(n-1) + \alpha \times M^{**}(b-x)) \bmod (L \times M - 1) \dots (3)$$

$$\text{また、} REG = \alpha \times M^{**}(b-x) \bmod (L \times M - 1)$$

ただし、 $Ab(0) = 0$ 、 α は2以上の整数とする。また、**はべき乗を表わす。

【0047】

従って、上述の例では $\alpha = 2$ として最初の書き込みの際に1つ飛ばしに書き込みを行っており、 α の値を適宜設定することにより、2つ飛ばし以上で書き込みを行うことも可能であるが、 α と $L \times M - 1$ の間に公約数が存在してはいけない。これは、 α と $L \times M - 1$ の間に公約数が存在すると、ブロック内のデータの最後のデータがアドレス $L \times M - 1$ に書き込まれるはずなのに、途中でアドレスが

$L \times M - 1$ となり、アドレスの生成規則が破綻するからである。

また、 α が M の $(-X)$ 乗に等しい場合も除く必要がある。この場合は従来例に一致し、さらなる回路規模の縮小や消費電力の低減が実現できないからである。

【0048】

次にこのような書き込み読み出し動作を行うのに必要な、アドレス生成装置のアドレス生成動作について説明する。

図1のアドレス生成装置103において、定数発生器118は初期値「 α 」を発生し、これをレジスタ113に出力する。乗算器111はこのレジスタ113の出力と定数発生器110の出力「 M 」を乗算し、オーバフロー処理装置140に出力する。

【0049】

オーバフロー処理装置140はその入力データが $L \times M - 1$ 以下になるまで「 $L \times M - 1$ 」を減算し、その結果をレジスタ113に出力する。このレジスタ113は、ブロックの先頭入力データ同期信号114により入力データが $L \times M$ 個入力された時点で初めてオーバフロー処理装置140の出力値に更新される。

【0050】

また、定数発生器119は初期値「1」を発生し、これをレジスタ117に出力する。加算器115はこのレジスタ117の出力とレジスタ113の出力を加算し、オーバフロー処理装置141に出力する。

【0051】

オーバフロー処理装置141はその入力データが $L \times M - 1$ 以下になるまで「 $L \times M - 1$ 」を減算し、その結果をレジスタ117に出力する。このレジスタ117は、ブロックの先頭入力データ同期信号114により入力データが $L \times M$ 個入力された時点で初期値「1」にリセットされ、同期信号入力106により入力データ毎に更新される。

【0052】

これにより、アドレス生成装置において、ブロック番号 b の 0 番目のアドレス $Ab(0)$ を 0 とし、ブロック番号 b の n 番目 (n は 0 以上の整数) のアドレス $Ab(n)$ を、 α (α は、 2 以上の整数) と M の $(b-x)$ 乗 (x は 0 以上 b 以下の整数)

の乗算結果と $Ab(n-1)$ との和を $L \times M-1$ で割った剰余から生成することにより、本実施の形態1の装置における記憶装置のアドレスの生成を実行しているものであり、かつオーバーフロー処理装置を設けることにより、乗算や加算を繰り返すことにより、アドレス生成装置内部で数値が $L \times M-1$ 以上に発散してゆくのを抑えている。

【0053】

図4はこのオーバーフロー処理装置140および141の構成を示している。図4(a)のオーバーフロー処理装置140において、121はスイッチであり、ブロックの先頭入力データ同期信号114に応じて本オーバーフロー処理装置の入力とレジスタ125の出力を切り替えて出力する。123はその入力が $L \times M-1$ 以上か否かを判定する比較器、122はその入力より $L \times M-1$ を減算する減算器、124はスイッチであり、比較器123の判定結果に応じてこれに入力されるスイッチ121の出力および減算器122の出力を切り替えて出力する。125はこのスイッチ124の出力を入力とするレジスタである。

【0054】

また、図4(b)のオーバーフロー処理装置141において、131はスイッチであり、ブロックの先頭入力データ同期信号114に応じて本オーバーフロー処理装置の入力とレジスタ135の出力を切り替えて出力する。133はその入力が $L \times M-1$ 以上か否かを判定する比較器、132はその入力より $L \times M-1$ を減算する減算器、134はスイッチであり、比較器133の判定結果に応じてこれに入力されるスイッチ131の出力および減算器132の出力を切り替えて出力する。135はこのスイッチ134の出力を入力とするレジスタである。

【0055】

次に動作について説明する。オーバーフロー処理装置140、141の動作は同様であるので、オーバーフロー処理装置140についてのみこれを述べる。まず、スイッチ121には本オーバーフロー処理装置140の入力およびレジスタ125の出力が与えられるが、入力データがブロックの先頭に該当し、ブロックの先頭入力データ同期信号114が入力されると、スイッチ121はオーバーフロー処理装置140の入力を選択し、それ以外の場合はレジスタ125の出力を選択する

。このスイッチ 121 の出力は比較器 123 により $L \times M - 1$ と比較される。スイッチ 124 にはスイッチ 121 の出力から $L \times M - 1$ を減算する減算器 122 の出力およびスイッチ 121 の出力が与えられるが、スイッチ 121 の出力が $L \times M - 1$ 以上である旨を比較器 123 が判定すると減算器 122 の出力を選択し、それ以外の場合はスイッチ 121 の出力を選択する。そしてこのスイッチ 124 の出力はレジスタ 125 に出力され、入力データ同期信号 106 により入力データ毎に更新される。これにより、オーバーフロー処理装置 140 は $L \times M - 1$ 以上であれば、 $L \times M - 1$ を繰り返し減算することにより、入力を $L \times M - 1$ 以下となるように制限する。

このようなオーバーフロー処理装置を設けることにより、アドレス生成装置の内部で乗算や加算を繰り返すことにより、数値が $L \times M - 1$ 以上に発散してゆくのを抑えている。

【0056】

また、式 (3) において、 x の値を設定することにより、初期状態を図 2 (a) 以外のいずれかの状態に変更することもできるが、この場合も上述の処理を繰り返すことにより、初期状態のブロックに戻り、以降の処理は同様の繰り返しとなる。

【0057】

このように、本実施の形態 1 は 1 ブロックの記憶領域を持つ記憶装置でブロックインタリーブを行えるものであり、この点は従来例と同様であるが、本実施の形態 1 はさらにそのアドレス生成装置の回路面積を削減できるものである。

【0058】

以下、この点につき説明する。

表 1 は本実施の形態 1 と同様の回路構成で従来例を構成した場合 (図 1 において、定数発生器 118 の α の値を 1 としたのが従来例であり、これを 2 以上とすれば本実施の形態 1 になる) に、レジスタ 113 の値の遷移を示したものである。

【0059】

【表1】

1							
2							
3	L=	4					
4	M=	5					
5	α =	1					
6							
7							
8	val=	1 →	5				
9	val=	5 →	25	6			
10	val=	6 →	30	11			
11	val=	11 →	55	36	17		
12	val=	17 →	85	66	47	28	9
13	val=	9 →	45	26	7		
14	val=	7 →	35	16			
15	val=	16 →	80	61	42	23	4
16	val=	4 →	20	1			
17							
18	overtime	=	16				
19	maxoverval	=	85				
20	minoverval	=	20				
21	maxval	=	17				
22							
23							
24	L=	4					
25	M=	5					
26	α =	2					
27							
28							
29	val=	2 →	10				
30	val=	10 →	50	31	12		
31	val=	12 →	60	41	22	3	
32	val=	3 →	15				
33	val=	15 →	75	56	37	18	
34	val=	18 →	90	71	52	33	14
35	val=	14 →	70	51	32	13	
36	val=	13 →	65	46	27	8	
37	val=	8 →	40	21	2		
38							
39	overtime	=	20				
40	maxoverval	=	90				
41	minoverval	=	21				
42	maxval	=	18				

【0060】

この表1はL=4、M=5、即ち4行5列のデータに対しブロックインタリーブを行う場合のレジスタ113の値の推移を示したものであり、表1中のvalとして示したのがレジスタ113の値であり、しきい値19（=5×4-1）を越える場合はオーバーフロー処理装置によってこのしきい値内に収まるように、順次値が減少するように処理されている。

【0061】

また、overtimeはレジスタ113の値がしきい値を越える回数、maxovervalはしきい値を越えるレジスタ113の値の最大値、minovervalはしきい値を越え

るレジスタ 113 の値の最小値、maxval はレジスタ 113 の値の最大値である

【0062】

また、この表 1 の行 8 ないし行 16 は従来例（行 5 で $\alpha = 1$ と設定）におけるレジスタ 113 の値の推移を示しており、行 29 ないし行 37 は本実施の形態 1（行 26 で $\alpha = 2$ と設定）におけるレジスタ 113 の値の推移を示している。

【0063】

これらを対比することにより、従来例ではしきい値を越えるレジスタ 113 の値の最小値 minoverval が 20（ $= L \times M$ の値、即ちしきい値 19 を越える最小の値）であるのに対し、本実施の形態 1 では 21、すなわち従来例よりも大きくなっていることが分かる。

【0064】

また表 2 の行 3 ないし行 21 は $L = 8$ 、 $M = 203$ 、即ち 8 行 203 列のデータに対しブロックインタリーブを行う場合のレジスタ 113 の値の計算結果を示したものであり、表 2 の行 8 ないし 11 が従来例のレジスタ 113 の値の計算結果を示したものである。また表 2 の行 18 ないし行 21 が本実施の形態 1 におけるレジスタ 113 の値の計算結果を示したものである。

【0065】

【表 2】

1	
2	
3	-----
4	L= 8
5	M= 203
6	$\alpha = 1$
7	-----
8	overtime = 16362
9	maxoverval = 325409
10	minoverval = 1624
11	maxval = 1603
12	
13	-----
14	L= 8
15	M= 203
16	$\alpha = 20$
17	-----
18	overtime = 19998
19	maxoverval = 329266
20	minoverval = 1643
21	maxval = 1622
22	

【0066】

これらを対比することにより、従来例ではオーバーフロー処理装置 140 のしきい値を越えるレジスタ 113 の値の最小値 minoverval が 1624 ($=L \times M$ の値、即ちしきい値 1623 を越える最小の値) であるのに対し、本実施の形態 1 では 1643、すなわち従来例よりも大きくなっていることが分かる。

【0067】

このように、本実施の形態 1 では、記憶装置に書き込み、読み出しを行う際の最初の書き込みにおいて、従来例ではアドレス通りに順次書き込みを行っているのに対し、1つ以上飛ばして書き込みを行うようにしており、この最初の書き込み順序が異なることにより、レジスタ 113 に保持される、しきい値を越える最小の値が従来例と同一かそれより大きくなる。

【0068】

これにより、従来例では 1624 以上を比較する比較器が必要であったが、本実施の形態 1 では 1643 以上を比較する比較器でよいため比較器の機能が簡略化する。

このように、オーバーフロー処理装置内の比較器で入力と比較すべきしきい値を $L \times M$ よりも大きくできる場合は、比較器の回路規模を従来例より必ず小さくできる。

【0069】

以下、この点を、8行203列のデータに対しブロックインタリーブを行う装置を例にとって説明する。

この場合、従来例の方式では、オーバーフロー処理装置 140 内の比較器 123 は入力が $L \times M$ 、即ち 1624 以上であることを判定しなければならない。

図5は従来例の方式により8行203列のデータに対しブロックインタリーブを行う装置のオーバーフロー処理装置における比較器の構成を示すものである。

図5において、3311ないし3319および3321ないし3333はANDゲート、3336ないし3339および3350ないし3356はORゲートである。

【0070】

次に動作について説明する。入力Iが1624以上である旨を判定するには、入力Iのビットパターンがこの1624を2進数に展開した011001011000以上であることを判定すればよい。そしてその際、入力Iの下位3ビットはその値が0であっても1であっても判定に支障はなく、これら下位3ビットが全て1の場合の入力値は1631になる。従って、入力値が1624である旨を判定する際に下位3ビットを入力しないことにより、入力値が1624～1631である旨を判定することができる。

【0071】

ANDゲート3311ないし3319はこうした原理により入力値が1624～1631である旨を判定するものであり、ANDゲート3311ないし3314により入力値の12ビット目ないし5ビット目のビットパターンが01100101100に一致する場合にそれぞれ1を出力する。ANDゲート3315および3316はこれら4つのANDゲート3311ないし3314の出力が全て1である場合にそれぞれ1を出力し、ANDゲート3317はANDゲート3315および3316の出力が全て1である場合に1を出力する。また、ANDゲート3318は入力値の4ビット目が1であり、かつANDゲート3316の出力が1である場合に1を出力する。さらに、ANDゲート3319はANDゲート3317および3318の出力が全て1である場合に1を出力する。従って、ANDゲート3319の出力が1の場合、入力値が1624～1631である旨が判明する。

【0072】

同様に、ANDゲート3321ないし3326は入力が1632～1663である旨を判定する。また、ANDゲート3327ないし3330は入力が1664～1791である旨を判定する。また、ANDゲート3331ないし3333は入力が1792～2047である旨を判定する。また、ORゲート3350ないし3356は入力が2048～524287 (maxovervalが325409であるため524287まで判定する。)である旨を判定する。

従って、ORゲート3336ないし3339によりこれらの判定結果を束ねる

ことにより、入力値が1624以上である旨を判定することができる。

【0073】

このように、従来例では比較器は入力が $L \times M$ 、即ち1624以上であることを判定しなければならないが、表2の行1ないし行11及び表2の行13ないし行21の対比により、本実施の形態1では入力が1643以上である旨を判定すればよい。

【0074】

図6は本実施の形態1のブロックインタリーブを行う装置のオーバフロー処理装置における比較器の構成を示すものである。

図6において、3321ないし3333はANDゲート、3340ないし3342および3350ないし3356はORゲートである。

この図6では、本来入力が1643以上である旨を判定すべきであるが、この判定は、1632以上を判定する場合に含まれるため、この回路では、1632以上である旨を判定している。

【0075】

まず、入力が1632～1663である旨をANDゲート3321ないし3326が判定する。また、入力が1664～1791である旨をANDゲート3327ないし3330が判定する。また、入力が1792～2047である旨をANDゲート3331ないし3333が判定する。また、入力が2048～524287（maxovervalが329266であるため524287まで判定する。）である旨をORゲート3350ないし3356が判定する。

従って、ORゲート3340ないし3342によりこれらの判定結果を束ねることにより、入力値が1632以上、即ち1643以上である旨を判定することができる。

【0076】

この図6の回路はANDゲートが13個、ORゲートが10個必要であるが、従来例に対応する図5の回路では、ANDゲートが22個、ORゲートが11個必要であり、この図6の回路は、比較すべき対象が減った分、従来例よりも回路規模を縮小でき、省面積化が図れ、かつその分消費電力も低減することができる。

【0077】

ところで、この $L=8$ 、 $M=203$ 、 $\alpha=20$ の場合は、BSデジタル放送における誤り訂正に用いて有効なものである。

即ち、BSデジタル放送の場合、リードソロモン復号器の訂正の対象となる1データセグメントは、データインタリーブ装置の内では203byteとなっており、送信側のブロックインタリーブ装置の列の数が203であれば、もっとも少ないインタリーブ装置の記憶容量でリードソロモン復号器の訂正能力を向上することが可能となる。また、行および列の数が増加するほど、連続したバースト誤りに対してのリードソロモン復号器の訂正能力が向上する。

【0078】

なお、 α は $L \times M - 1$ との間に公約数が存在せず、 M の $(-X)$ 乗に等しくなければ、2以上の任意の整数であってよいが、 $\alpha=20$ の場合がもっとも効果大きい。

また、以上の場合とは異なる原理により消費電力を低減できる場合がある。

以下、この場合について述べると、表3は $L=10$ 、 $M=8$ 、即ち10行8列のデータに対しブロックインタリーブを行う場合のレジスタ113の値の推移を、本実施の形態1と従来例の回路について対比して示したものである。

【0079】

【表 3】

1										
2	-----									
3	L=	10								
4	M=	8								
5	α =	1								
6	-----									
7										
8	val=	1 →	10							
9	val=	10 →	100	21						
10	val=	21 →	210	131	52					
11	val=	52 →	520	441	362	283	204	125	46	
12	val=	46 →	460	381	302	223	144	65		
13	val=	65 →	650	571	492	413	334	255	176	97 18
14	val=	18 →	180	101	22					
15	val=	22 →	220	141	62					
16	val=	62 →	620	541	462	383	304	255	146	67
17	val=	67 →	670	591	512	433	354	275	196	117 38
18	val=	38 →	380	301	222	143	64			
19	val=	64 →	640	561	482	403	324	245	166	87 8
20	val=	8 →	80	1						
21										
22	overtime	=	54							
23	maxoverval	=	670							
24	minoverval	=	80							
25	maxval	=	67							
26										
27	-----									
28	L=	10								
29	M=	8								
30	α =	4								
31	-----									
32										
33	val=	4 →	40							
34	val=	40 →	400	321	242	163	84	5		
35	val=	5 →	50							
36	val=	50 →	500	421	342	263	184	105	26	
37	val=	26 →	260	181	102	23				
38	val=	23 →	230	151	72					
39	val=	72 →	720	641	562	483	404	325	246	167 88 9
40	val=	9 →	90	11						
41	val=	11 →	110	31						
42	val=	31 →	310	231	152	73				
43	val=	73 →	730	651	572	493	414	335	256	177 98 19
44	val=	19 →	190	111	32					
45	val=	32 →	320	241	162	83	4			
46										
47	overtime	=	45							
48	maxoverval	=	730							
49	minoverval	=	83							
50	maxval	=	73							

【 0 0 8 0 】

この表 3 より明らかなように、従来例ではレジスタ 1 1 3 の値がしきい値を越える回数overtimeが 5 4 回であるのに対し、本実施の形態 1 ではこれが 4 5 回に減少しており、このovertimeの減少により、オーバーフロー処理装置 1 4 0 における計算量が減少し、さらに従来の方式ではオーバーフロー処理装置 1 4 1 のオーバーフロー回数が 4 7 4 回であったが、本実施の形態 1 では 3 9 5 回に減少すること

によりオーバーフロー処理装置 141 の計算量が減少する。

これらにより消費電力の減少が実現できる。

【0081】

このように、本実施の形態 1 によるブロックインタリーブ装置は、ブロックインタリーブ装置出力を生成する $L \times M$ データの記憶装置と前記記憶装置にアドレスを出力するアドレス生成装置と前記記憶装置へ制御信号を出力する記憶装置制御装置から構成され、アドレス生成装置において、ブロック番号 b の 0 番目のアドレス $Ab(0)$ を 0 とし、ブロック番号 b の n 番目 (n は 0 以上の整数) のアドレス $Ab(n)$ を、 α (α は、2 以上の整数) と M の $(b-x)$ 乗 (x は 0 以上 b 以下の整数) の乗算結果と $Ab(n-1)$ との加算結果を $L \times M - 1$ で割った剰余から生成し、生成したアドレスに対して読み出し、書き込みを繰り返すことによりブロックインタリーブを行うようにしたので、記憶装置アドレス生成装置を最適化でき、最小限の回路面積により、ブロックインタリーブを行うことが可能となる。

【0082】

また、ブロックの先頭アドレスと最終アドレスが常に一定であるので、これらに対し記憶装置内の連続した領域に割り当てることにより、2 データについて同時に処理することが可能となり、記憶装置のアクセス回数が低減し、アドレス生成装置の低消費電力化が可能となる。

【0083】

また、特に、 $L=8$ 、 $M=203$ のブロックインタリーブを行う場合、特表平 8-511393 号公報記載の、従来のアドレス生成装置においては、ブロック番号 b の 0 番目のアドレス $Ab(0)$ を 0 とし、ブロック番号 b の n 番目 (n は 0 以上の整数) のアドレス $Ab(n)$ を、 M の $(b-x)$ 乗 (x は 0 以上 b 以下の整数) と $Ab(n-1)$ との加算結果を $L \times M - 1$ で割った剰余から生成する。この演算を繰り返していくと、剰余の対象となる値が無限に大きくなるため、回路で実現する場合、初期値を M の $(b-x-1)$ 乗とし入力に M を乗算してオーバーフロー処理装置 1 (以下、剰余器 1 と称す) へ出力する乗算器と、入力を $L \times M - 1$ で割った剰余を乗算器と加算器に出力する剰余器 1 と $Ab(n-1)$ と剰余器 1 の出力を加算してオーバーフロー処理装置 2 (以下、剰余器 2 と記す) へ出力する加算器と入力を $L \times M - 1$ で割った剰余

余を $Ab(n)$ とする剰余器2から構成されることになるが、剰余器1は入力 $L \times M-1$ 以下となるまで、 $L \times M-1$ を減算するための、比較器と減算器から構成され、減算の対象となる最小の値は、『1624』となり、比較器は『1624』以上を判定する機能が必要となる。

【0084】

しかしながら、本実施の形態1によるブロックインタリーブ装置において、 $\alpha = 20$ 、 $L = 8$ 、 $M = 203$ とすると、初期値を M の $(b-x-1)$ 乗に α を乗算したものとし入力に M を乗算して剰余器1へ出力する乗算器と入力を $L \times M-1$ で割った剰余を乗算器と加算器に出力する剰余器1と $Ab(n-1)$ と剰余器1の出力を加算して剰余器2へ出力する加算器と入力を $L \times M-1$ で割った剰余を $Ab(n)$ とする剰余器2から構成され、剰余器1は入力 $L \times M-1$ 以下となるまで、 $L \times M-1$ を減算するための、比較器と減算器から構成され、減算の対象となる最小の値は『1643』となり、比較器は、『1643』以上を判定する機能だけでよいことから、比較器の面積が低減し、最小限の回路面積でブロックインタリーブを行なうことが可能となる。

【0085】

なお、読み出しアドレスを $Ab(n)$ とし、書き込みアドレスを $Ab(n-t)$ 、(t は $L \times M-2$ 以下の自然数)とし、それぞれのアドレスに対し、1時点ごとに、読み出しと書き込みを繰り返すことにより、ブロックインタリーブを実現することも可能である。

また、 $Ab(0)$ を β とし(β は、 $L \times M-1$ 以下の自然数)、 $Ab(n)$ を α と M の $(b-x)$ 乗の乗算結果と $Ab(n-1)$ と β との加算結果を $L \times M-1$ で割った剰余から生成することも可能である。

【0086】

(実施の形態2)

以下、本発明の実施の形態2について、図を用いて説明する。

本願発明によるブロックデインタリーブ装置およびブロックデインタリーブ方法について説明する。

本実施の形態2のブロックデインタリーブ装置およびブロックデインタリーブ

方法は、記憶装置のアドレス生成装置を最適化することにより、記憶装置の制御回路の面積あるいは消費電力をより低減できるようにしたものである。

【0087】

図7は本願発明の実施の形態2による、 $L \times M$ データのブロックデインタリーブを行うブロックデインタリーブ装置を示すものである。図7において、1は本ブロックデインタリーブ装置によりブロックデインタリーブを行うべき入力データの入力端子、14はこのブロックデインタリーブを行うべき入力データのブロックの先頭入力データ毎に同期して入力される先頭入力データ同期信号の入力端子、6は各入力データ毎に発生する同期信号の入力端子、12は同期信号入力端子6から入力される同期信号に応じて記憶装置4を制御する制御装置、3は入力端子6から入力される同期信号および入力信号14から入力される先頭入力データ同期信号に基づいて記憶装置4のアドレスを生成するアドレス生成装置、20はこのアドレス生成装置3が生成したアドレスを出力する出力端子、4は記憶装置であり、制御装置12の制御により、アドレス生成装置3により生成されたアドレスに入力端子1からの入力データを書き込み、これを読み出すことによりブロックデインタリーブを行うものである。5は本ブロックデインタリーブ装置によりデインタリーブされたデータを出力する出力端子である。

【0088】

また、図7のアドレス生成装置3において、10は初期値 L を発生する定数発生器、11はこの初期値 L とレジスタ13の出力を乗算する乗算器、40はこの乗算器11の出力がオーバーフローした場合の処理を行うオーバーフロー処理装置、13はこのオーバーフロー処理装置40の出力または定数発生器18の出力がセットされるレジスタ、18は初期値 α を発生する定数発生器、15はこのレジスタ13の出力とレジスタ17の出力を加算する加算器、41はこの加算器15の出力がオーバーフローした場合の処理を行うオーバーフロー処理装置、17はこのオーバーフロー処理装置41の出力または定数発生器19の出力がセットされるレジスタ、19は初期値1を発生する定数発生器である。

図8は本実施の形態2によるブロックデインタリーブ装置の動作を模式的に示すものであり、4行5列のデータに対し、ブロックデインタリーブを行う場合を

例にとって示している。

【0089】

次に図7の動作について、図8に示した、4行5列のデータに対し、ブロックデインタリーブを行う場合を例にとって説明する。

図8に示すように、本実施の形態2によるブロックデインタリーブ装置は、入力端子1から入力されるデータを $L \times M$ データの記憶装置4に書き込み、この $L \times M$ データの記憶装置4からデータを読み出すことによりブロックデインタリーブを行うものであり、その際、図8に示すような順序で書き込み、読み出しを行うように、制御装置12が前記記憶装置4へ制御信号を出力することによりその書き込み、読み出し制御を行い、かつアドレス生成装置3がその書き込み、読み出しの際のアドレスを生成し、これを前記記憶装置4に出力することにより、1ブロックの記憶領域を持つ記憶装置1面でブロックデインタリーブされた出力5を生成できるものである。

【0090】

この装置は、ブロックデインタリーブ装置の記憶装置4のアドレスが図13(a)のように割り当てられているとすると、まず、図8(a)のようにREGを2とし、図13(a)のアドレス0を初期値とし入力データ毎に2増加する書き込みアドレスを順次生成する。その際、書き込みアドレスが19($=4 \times 5 - 1$)を越えるところの19を減算する。そして、この生成規則に従って生成した書き込みアドレスに従い、データを書き込みを、ブロック内の全てのアドレスにアクセスが完了する迄行なう。

即ち、図13(a)の従来方式では書き込みアドレスが増加する順に従って0→1→2→……→19の順でデータを順次書き込んでいたものが、本実施の形態2ではこれを1つ飛ばしに書き込んでゆくものである。

【0091】

次に、図8(b)に示されているように、REGに4を乗じ、図13(a)におけるアドレスの並びを基準として、図13(a)のアドレス0を初期値とし、1入力データ毎にアドレスが8($=2 \times 4$)増加するアドレスを順次生成するが、その際、アドレスが19($=4 \times 5 - 1$)を越えるところの19を減算する。

そして、この生成規則に従って生成したアドレスに従い読み出しおよび書き込みを、ブロック内の全てのアドレスにアクセスが完了する迄行なう。

【0092】

次に、図8(c)に示されているように、このREGに対し4を乗じるが、その値が19を越えているのでこの19により除算を行いその剰余13を求める。

そして、図13(a)に示されているアドレスの並びを基準とし、そのアドレス0を初期値として、1入力データ毎に13増加するアドレスを順次生成し、アドレスが19(=4×5-1)を越えとこの19を減算する。

そして、この生成規則に従って生成したアドレスに従い読み出しおよび書き込みを、ブロック内の全てのアドレスにアクセスが完了する迄行なう。

【0093】

以下、同様の処理を繰り返すことにより、順次異なるアドレスの順で読み出し、書き込みを行ってゆけば、この例では図8(j)の時点において、図8(a)と同様のアドレスの順に戻る。

このような手順を繰り返すことにより、図9に示すように、1ブロックの記憶領域を持つ記憶装置1面を用いてブロックデインタリーブを行うことを可能としている。これは制御装置12による書き込み読み出し制御、およびアドレス生成装置3により生成する記憶装置4のアドレスを上述のように工夫することにより実現しているものであり、これに加えて、本実施の形態2ではアドレス生成装置の回路面積や消費電力の低減が可能になっているものである。

【0094】

この実施の形態2におけるアドレス生成規則は以下の通りである。

即ち、n番目のアドレスを $Ab(n)$ 、記憶装置の行の数をL、列の数をM、ブロック番号bを0以上の整数、xを0以上b以下の任意の整数とすると、

$$Ab(n) = (Ab(n-1) + \alpha \times L * (b-x)) \bmod (L \times M - 1) \dots$$

(4)

また、 $REG = \alpha \times L * (b-x) \bmod (L \times M - 1)$

ただし、 $Ab(0) = 0$ 、 α は2以上の整数とする。また、 $*$ はべき乗を表わす。

【0095】

従って、上述の例では $\alpha = 2$ として最初の書き込みの際に1つ飛ばしに書き込みを行っており、 α の値を適宜設定することにより、2つ飛ばし以上で書き込みを行うことも可能であるが、 α と $L \times M - 1$ の間に公約数が存在してはいけない。これは、 α と $L \times M - 1$ の間に公約数が存在すると、ブロック内のデータの最後のデータがアドレス $L \times M - 1$ に書き込まれるはずなのに、途中でアドレスが $L \times M - 1$ となり、アドレスの生成規則が破綻するからである。

また、 α が L の(-X)乗に等しい場合も除く必要がある。この場合は従来例に一致し、さらなる回路規模の縮小や消費電力の低減が実現できないからである。

【0096】

次にこのような書き込み読み出し動作を行うのに必要な、アドレス生成装置のアドレス生成動作について説明する。

図7のアドレス生成装置3において、定数発生器18は初期値「 α 」を発生し、これをレジスタ13に出力する。乗算器11はこのレジスタ13の出力と定数発生器10の出力「 L 」を乗算し、オーバフロー処理装置40に出力する。

【0097】

オーバフロー処理装置40はその入力データが $L \times M - 1$ 以下になるまで「 $L \times M - 1$ 」を減算し、その結果をレジスタ13に出力する。このレジスタ13は、ブロックの先頭入力データ同期信号14により入力データが $L \times M$ 個入力された時点で初めてオーバフロー処理装置40の出力値に更新される。

【0098】

また、定数発生器19は初期値「1」を発生し、これをレジスタ17に出力する。加算器15はこのレジスタ17の出力とレジスタ13の出力を加算し、オーバフロー処理装置41に出力する。

オーバフロー処理装置41はその入力データが $L \times M - 1$ 以下になるまで「 $L \times M - 1$ 」を減算し、その結果をレジスタ17に出力する。このレジスタ17は、ブロックの先頭入力データ同期信号14により入力データが $L \times M$ 個入力された時点で初期値「1」にリセットされ、同期信号入力6により入力データ毎に更新される。

【0099】

これにより、アドレス生成装置において、ブロック番号 b の0番目のアドレス $Ab(0)$ を0とし、ブロック番号 b の n 番目(n は0以上の整数)のアドレス $Ab(n)$ を、 α (α は、2以上の整数)と L の $(b-x)$ 乗(x は0以上 b 以下の整数)の乗算結果と $Ab(n-1)$ との和を $L \times M-1$ で割った剰余から生成することにより、本実施の形態2の装置における記憶装置のアドレスの生成を実行しているものであり、かつオーバーフロー処理装置を設けることにより、乗算や加算を繰り返すことにより、アドレス生成装置内部で数値が $L \times M-1$ 以上に発散してゆくのを抑えている。

【0100】

図10はこのオーバーフロー処理装置40および41の構成を示している。図10(a)のオーバーフロー処理装置40において、21はスイッチであり、ブロックの先頭入力データ同期信号14に応じて本オーバーフロー処理装置の入力とレジスタ25の出力を切り替えて出力する。23はその入力が $L \times M-1$ 以上か否かを判定する比較器、22はその入力より $L \times M-1$ を減算する減算器、24はスイッチであり、比較器23の判定結果に応じてこれに入力されるスイッチ21の出力および減算器22の出力を切り替えて出力する。25はこのスイッチ24の出力を入力とするレジスタである。

【0101】

また、図10(b)のオーバーフロー処理装置41において、31はスイッチであり、ブロックの先頭入力データ同期信号14に応じて本オーバーフロー処理装置の入力とレジスタ35の出力を切り替えて出力する。33はその入力が $L \times M-1$ 以上か否かを判定する比較器、32はその入力より $L \times M-1$ を減算する減算器、34はスイッチであり、比較器33の判定結果に応じてこれに入力されるスイッチ31の出力および減算器32の出力を切り替えて出力する。35はこのスイッチ34の出力を入力とするレジスタである。

【0102】

次に動作について説明する。オーバーフロー処理装置40、41の動作は同様であるので、オーバーフロー処理装置40についてのみこれを述べる。まず、スイッ

チ 21 には本オーバフロー処理装置 40 の入力およびレジスタ 25 の出力が与えられるが、入力データがブロックの先頭に該当し、ブロックの先頭入力データ同期信号 14 が入力されると、スイッチ 21 はオーバフロー処理装置 40 の入力を選択し、それ以外の場合はレジスタ 25 の出力を選択する。このスイッチ 21 の出力は比較器 23 により $L \times M - 1$ と比較される。スイッチ 24 にはスイッチ 21 の出力から $L \times M - 1$ を減算する減算器 22 の出力およびスイッチ 21 の出力が与えられるが、スイッチ 21 の出力が $L \times M - 1$ 以上である旨を比較器 23 が判定すると減算器 22 の出力を選択し、それ以外の場合はスイッチ 21 の出力を選択する。そしてこのスイッチ 24 の出力はレジスタ 25 に出力され、入力データ同期信号 6 により入力データ毎に更新される。これにより、オーバフロー処理装置 40 は $L \times M - 1$ 以上であれば、 $L \times M - 1$ を繰り返し減算することにより、入力を $L \times M - 1$ 以下となるように制限する。

このようなオーバフロー処理装置を設けることにより、アドレス生成装置の内部で乗算や加算を繰り返すことにより、数値が $L \times M - 1$ 以上に発散してゆくのを抑えている。

【0103】

また、式(4)において、 x の値を設定することにより、初期状態を図 8(a) 以外のいずれかの状態に変更することもできるが、この場合も上述の処理を繰り返すことにより、初期状態のブロックに戻り、以降の処理は同様の繰り返しとなる。

このように、本実施の形態 2 は 1 ブロックの記憶領域を持つ記憶装置でブロックデインタリーブを行えるものであり、この点は従来例と同様であるが、本実施の形態 2 はさらにそのアドレス生成装置の回路面積を削減できるものである。

【0104】

以下、この点につき説明する。

表 4 の行 1 ないし行 21 は $L = 8$, $M = 203$ 、即ち 8 行 203 列のデータに対しブロックデインタリーブを行う場合のレジスタ 13 の値の計算結果を示したものであり、表 4 の行 8 ないし 11 が従来例のレジスタ 13 の値の計算結果を示したものである。また表 4 の行 18 ないし行 21 が本実施の形態 2 におけるレジ

スタ 13 の値の計算結果を示したものである。

【0105】

【表 4】

1		
2		
3	-----	
4	L= 8	
5	M= 203	
6	α = 1	
7	-----	
8	overtime =	567
9	maxoverval =	12824
10	minoverval =	1624
11	maxval =	1603
12		
13	-----	
14	L= 8	
15	M= 203	
16	α = 20	
17	-----	
18	overtime =	693
19	maxoverval =	12967
20	minoverval =	1643
21	maxval =	1622
22		

【0106】

これらを対比することにより、従来例ではオーバフロー処理装置 40 のしきい値を越えるレジスタ 13 の値の最小値 minoverval が 1624 (= L × M の値、即ちしきい値 1623 を越える最小の値) であるのに対し、本実施の形態 2 では 1643、すなわち従来例よりも大きくなっていることが分かる。

【0107】

このように、本実施の形態 2 では、記憶装置に書き込み、読み出しを行う際の最初の書き込みにおいて、従来例ではアドレス通りに順次書き込みを行っているのに対し、1つ以上飛ばして書き込みを行うようにしており、この最初の書き込み順序が異なることにより、レジスタ 13 に保持される、しきい値を越える最小の値が従来例と同一かそれより大きくなる。

【0108】

これにより、従来例では 1624 以上を比較する比較器が必要であったが、本実施の形態 1 では 1643 以上を比較する比較器でよいため比較器の機能が簡略化する。

このように、オーバフロー処理装置内の比較器で入力と比較すべきしきい値を

L×Mより大きくできる場合は、比較器の回路規模を従来例よりも必ず小さくできる。

【0109】

以下、この点を、8行203列のデータに対しブロックデインタリーブを行う装置を例にとって説明する。

この場合、従来例の方式では、オーバーフロー処理装置40内の比較器23は入力がL×M、即ち1624以上であることを判定しなければならない。

【0110】

図11は従来例の方式により8行203列のデータに対しブロックデインタリーブを行う装置のオーバーフロー処理装置における比較器の構成を示すものである。

図11において、2311ないし2319および2321ないし2333はANDゲート、2334ないし2339はORゲートである。

【0111】

次に動作について説明する。入力が1624以上である旨を判定するには、入力Iのビットパターンがこの1624を2進数に展開した011001011000以上であることを判定すればよい。そしてその際、入力Iの下位3ビットはその値が0であっても1であっても判定に支障はなく、これら下位3ビットが全て1の場合の入力値は1631になる。従って、入力値が1624である旨を判定する際に下位3ビットを入力しないことにより、入力値が1624～1631である旨を判定することができる。

【0112】

ANDゲート2311ないし2319はこうした原理により入力値が1624～1631である旨を判定するものであり、ANDゲート2311ないし2314により入力値の12ビット目ないし5ビット目のビットパターンが01100101100に一致する場合にそれぞれ1を出力する。ANDゲート2315および2316はこれら4つのANDゲート2311ないし2314の出力が全て1である場合にそれぞれ1を出力し、ANDゲート2317はANDゲート2315および2316の出力が全て1である場合に1を出力する。また、ANDゲ

ート 2318 は入力値の 4 ビット目が 1 であり、かつ AND ゲート 2316 の出力が 1 である場合に 1 を出力する。さらに、AND ゲート 2319 は AND ゲート 2317 および 2318 の出力が全て 1 である場合に 1 を出力する。従って、AND ゲート 2319 の出力が 1 の場合、入力値が 1624 ~ 1631 である旨が判明する。

【0113】

同様に、AND ゲート 2321 ないし 2326 は入力が 1632 ~ 1663 である旨を判定する。また、AND ゲート 2327 ないし 2330 は入力が 1664 ~ 1791 である旨を判定する。また、AND ゲート 2331 ないし 2333 は入力が 1792 ~ 2047 である旨を判定する。また、OR ゲート 2334 および 2335 は入力が 2048 ~ 16383 (maxoverval が 12824 であるため 16383 まで判定する。) である旨を判定する。

従って、OR ゲート 2336 ないし 2339 によりこれらの判定結果を束ねることにより、入力値が 1624 以上である旨を判定することができる。

【0114】

このように、従来例では比較器は入力が $L \times M$ 、即ち 1624 以上であることを判定しなければならないが、表 4 の行 8 ないし行 11 および表 4 の行 18 ないし行 21 の対比により、本実施の形態 2 では入力が 1643 以上である旨を判定すればよい。

【0115】

図 12 は本実施の形態 2 のブロックデインタリーブを行う装置のオーバフロー処理装置における比較器の構成を示すものである。

図 12 において、2321 ないし 2333 は AND ゲート、2334、2335 および 2340 ないし 2342 は OR ゲートである。

この図 12 では、本来入力が 1643 以上である旨を判定すべきであるが、この判定は、1632 以上を判定する場合に含まれるため、この回路では、1632 以上である旨を判定している。

【0116】

まず、入力が 1632 ~ 1663 である旨を AND ゲート 2321 ないし 23

26が判定する。また、入力が1664～1791である旨をANDゲート2327ないし2330が判定する。また、入力が1792～2047である旨をANDゲート2331ないし2333が判定する。また、入力が2048～16383 (maxovervalが12967 であるため16383 まで判定する。) である旨をORゲート2334および2335が判定する。

【0117】

従って、ORゲート2340ないし2342によりこれらの判定結果を束ねることにより、入力値が1632以上、即ち1643以上である旨を判定することができる。

【0118】

この図12の回路はANDゲートが13個、ORゲートが5個必要であるが、従来例に対応する図11の回路では、ANDゲートが22個、ORゲートが6個必要であり、この図12の回路は、比較すべき対象が減った分、従来例よりも回路規模を縮小でき、省面積化が図れ、かつその分消費電力も低減することができる。

【0119】

ところで、この $L=8$ 、 $M=203$ 、 $\alpha=20$ の場合は、BSデジタル放送における誤り訂正に用いて有効なものである。

即ち、BSデジタル放送の場合、リードソロモン復号器の訂正の対象となる1データセグメントは、データインタリーブ装置の内では203byteとなっており、送信側のブロックインタリーブ装置の列の数が203であれば、もっとも少ないインタリーブ装置の記憶容量でリードソロモン復号器の訂正能力を向上することが可能となる。また、行および列の数が増加するほど、連続したバースト誤り、に対してのリードソロモン復号器の訂正能力が向上する。

【0120】

従って、これに対応する受信側のブロックデインタリーブ装置の側では、上述のように、 $L=8$ 、 $M=203$ 、 $\alpha=20$ に設定することにより、最小の回路規模でバースト誤りに対する訂正能力を向上させ得るブロックデインタリーブ装置を得ることが可能となる。なお、 α は $L \times M - 1$ との間に公約数が存在せず、 L

の $(-X)$ 乗に等しくなければ、2以上の任意の整数であってよいが、 $\alpha = 20$ の場合がもっとも効果大きい。

【0121】

このように、本実施の形態2によるブロックデインタリーブ装置は、ブロックデインタリーブ装置出力を生成する $L \times M$ データの記憶装置と前記記憶装置にアドレスを出力するアドレス生成装置と前記記憶装置へ制御信号を出力する記憶装置制御装置から構成され、アドレス生成装置において、ブロック番号 b の0番目のアドレス $Ab(0)$ を0とし、ブロック番号 b の n 番目(n は0以上の整数)のアドレス $Ab(n)$ を、 α (α は、2以上の整数)と L の $(b-x)$ 乗(x は0以上 b 以下の整数)の乗算結果と $Ab(n-1)$ との加算結果を $L \times M-1$ で割った剰余から生成し、生成したアドレスに対して読み出し、書き込みを繰り返すことによりブロックデインタリーブを行うようにしたので、記憶装置アドレス生成装置を最適化でき、最小限の回路面積により、ブロックデインタリーブを行うことが可能となる。

【0122】

また、ブロックの先頭アドレスと最終アドレスが常に一定であるので、これらに記憶装置内の連続した領域に割り当てることにより、2データについて同時に処理することが可能となり、記憶装置のアクセス回数が低減し、アドレス生成装置の低消費電力化が可能となる。

【0123】

また、特に、 $L=8$ 、 $M=203$ のブロックデインタリーブを行う場合、特表平8-511393号公報記載の、従来のアドレス生成装置においては、ブロック番号 b の0番目のアドレス $Ab(0)$ を0とし、ブロック番号 b の n 番目(n は0以上の整数)のアドレス $Ab(n)$ を、 L の $(b-x)$ 乗(x は0以上 b 以下の整数)と $Ab(n-1)$ との加算結果を $L \times M-1$ で割った剰余から生成する。この演算を繰り返していくと、剰余の対象となる値が無限に大きくなるため、回路で実現する場合、初期値を L の $(b-x-1)$ 乗とし入力に L を乗算してオーバーフロー処理装置1(以下、剰余器1と称す)へ出力する乗算器と、入力を $L \times M-1$ で割った剰余を乗算器と加算器に出力する剰余器1と $Ab(n-1)$ と剰余器1の出力を加算してオーバーフロー処

理装置 2（以下、剰余器 2 と記す）へ出力する加算器と入力を $L \times M - 1$ で割った剰余を $Ab(n)$ とする剰余器 2 から構成されることになるが、剰余器 1 は入力が $L \times M - 1$ 以下となるまで、 $L \times M - 1$ を減算するための、比較器と減算器から構成され、減算の対象となる最大の値は、「1624」となり、比較器は「1624」以上を判定する機能が必要となる。

【0 1 2 4】

しかしながら、本実施の形態 2 によるブロックデインタリーブ装置において、 $\alpha = 20$ 、 $L = 8$ 、 $M = 203$ とすると、初期値を L の $(b-x-1)$ 乗に α を乗算したものとし入力を L を乗算して剰余器 1 へ出力する乗算器と入力を $L \times M - 1$ で割った剰余を乗算器と加算器に出力する剰余器 1 と $Ab(n-1)$ と剰余器 1 の出力を加算して剰余器 2 へ出力する加算器と入力を $L \times M - 1$ で割った剰余を $Ab(n)$ とする剰余器 2 から構成され、剰余器 1 は入力が $L \times M - 1$ 以下となるまで、 $L \times M - 1$ を減算するための、比較器と減算器から構成され、減算の対象となる最大の値は「1643」となり、比較器は、「1643」以上を判定する機能だけでよいことから、比較器の面積が低減し、最小限のアドレス生成回路面積でブロックデインタリーブを行うことが可能となる。

【0 1 2 5】

なお、読み出しアドレスを $Ab(n)$ とし、書き込みアドレスを $Ab(n-t)$ 、 $(t$ は $L \times M - 2$ 以下の自然数) とし、それぞれのアドレスに対し、1 時点ごとに、読み出しと書き込みを繰り返すことにより、ブロックデインタリーブを実現することも可能である。

また、 $Ab(0)$ を β とし (β は、 $L \times M - 1$ 以下の自然数)、 $Ab(n)$ を α と L の $(b-x)$ 乗の乗算結果と $Ab(n-1)$ と β との加算結果を $L \times M - 1$ で割った剰余から生成することも可能である。

【0 1 2 6】

【発明の効果】

以上のように、本願の請求項 1 の発明に係るブロックインタリーブ装置によれば、アドレスが連続する順序で割り当てられた記憶装置に対し 1 アドレス以上の同一間隔でアドレスを飛ばして書き込みを行い、読み出しデータがブロックイン

タリープされたものとなるような順序で読み出しを行って出力データとし、その後この読み出しと同じ順序で書き込みを行うことによりブロックインタリープを行うようにしたので、1ブロックの記憶領域を持つ記憶装置1面でのブロックインタリープ動作が可能となり、かつ記憶装置制御装置の回路規模が縮小可能となる効果がある。

【0127】

また、本願の請求項2の発明に係るブロックインタリープ装置によれば、請求項1記載のブロックインタリープ装置において、本ブロックインタリープ装置の出力を生成するための $L \times M$ データの記憶装置と、前記記憶装置に出力するアドレスを生成するアドレス生成装置と、前記記憶装置に対し制御信号を出力する記憶装置制御装置とを備え、前記アドレス生成装置において、ブロック番号 b の0番目のアドレス $Ab(0)$ を0とし、ブロック番号 b の n 番目(n は0以上の整数)のアドレス $Ab(n)$ を、 α (α は、2以上の整数)と M の $(b-x)$ 乗(x は0以上 b 以下の整数)の乗算結果に $Ab(n-1)$ を加算した結果を $L \times M-1$ で割った剰余から生成し、この生成したアドレスに対して読み出し、書き込みを繰り返すことによりブロックインタリープを行うようにしたので、記憶装置、アドレス生成装置を最適化でき、最小限の回路面積により、ブロックインタリープを行うことが可能となる効果がある。

【0128】

また、本願の請求項3の発明に係るブロックインタリープ装置によれば、請求項2記載のブロックインタリープ装置において、前記 α と $L \times M-1$ との間に公約数が存在しないように、これらの値を設定したので、アドレスの生成規則が破綻するのを防止でき、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックインタリープを行うことが可能となる効果がある。

【0129】

また、本願の請求項4の発明に係るブロックインタリープ装置によれば、請求項2記載のブロックインタリープ装置において、前記 α と M の $(-x)$ 乗が等しくないように、これらの値を設定したので、最初の書き込みの時点においてアドレスを連続して書き込みを行うのを防止でき、記憶装置、アドレス生成装置を最適化

し、最小限の回路面積により、ブロックインタリーブを行うことが可能となる効果がある。

【0130】

また、本願の請求項5の発明に係るブロックインタリーブ装置によれば、請求項2記載のブロックインタリーブ装置において、前記 α 、 L 、 M の値を、それぞれ $\alpha=20$ 、 $L=8$ 、 $M=203$ としたので、アドレス生成装置を構成する比較器の面積を低減でき、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックインタリーブを行うことが可能となる効果がある。

【0131】

また、本願の請求項6の発明に係るブロックデインタリーブ装置によれば、アドレスが連続する順序で割り当てられた記憶装置に対し1アドレス以上の同一間隔でアドレスを飛ばして書き込みを行い、読み出しデータがブロックデインタリーブされたものとなるような順序で読み出しを行って出力データとし、その後この読み出しと同じ順序で書き込みを行うことによりブロックデインタリーブを行うようにしたので、1ブロックの記憶領域を持つ記憶装置1面でのブロックデインタリーブ動作が可能となり、かつ記憶装置制御装置の回路規模が縮小可能となる効果がある。

【0132】

また、本願の請求項7の発明に係るブロックデインタリーブ装置によれば、請求項6記載のブロックデインタリーブ装置において、本ブロックデインタリーブ装置の出力を生成するための $L \times M$ データの記憶装置と、前記記憶装置に出力するアドレスを生成するアドレス生成装置と、前記記憶装置に対し制御信号を出力する記憶装置制御装置とを備え、前記アドレス生成装置において、ブロック番号 b の0番目のアドレス $Ab(0)$ を0とし、ブロック番号 b の n 番目(n は0以上の整数)のアドレス $Ab(n)$ を、 α (α は、2以上の整数)と L の $(b-x)$ 乗(x は0以上 b 以下の整数)の乗算結果に $Ab(n-1)$ を加算した結果を $L \times M-1$ で割った剰余から生成し、この生成したアドレスに対して読み出し、書き込みを繰り返すことによりブロックデインタリーブを行うようにしたので、記憶装置、アドレス生成装置を最適化でき、最小限の回路面積により、ブロックデインタリーブを行

うことが可能となる効果がある。

【0133】

また、本願の請求項8の発明に係るブロックデインタリーブ装置によれば、請求項7記載のブロックデインタリーブ装置において、前記 α と $L \times M - 1$ との間に公約数が存在しないように、これらの値を設定したので、アドレスの生成規則が破綻するのを防止でき、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックデインタリーブを行うことが可能となる効果がある。

【0134】

また、本願の請求項9の発明に係るブロックデインタリーブ装置によれば、請求項7記載のブロックデインタリーブ装置において、前記 α と L の $(-x)$ 乗が等しくないように、これらの値を設定したので、最初の書き込みの時点においてアドレスを連続して書き込みを行うのを防止でき、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックデインタリーブを行うことが可能となる効果がある。

【0135】

また、本願の請求項10の発明に係るブロックデインタリーブ装置によれば、請求項7記載のブロックデインタリーブ装置において、前記 α 、 L 、 M の値を、それぞれ $\alpha=20$ 、 $L=8$ 、 $M=203$ としたので、アドレス生成装置を構成する比較器の面積を低減でき、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックデインタリーブを行うことが可能となる効果がある。

【0136】

また、本願の請求項11の発明に係るブロックインタリーブ方法によれば、アドレスが連続する順序で割り当てられた記憶装置に対し1アドレス以上の同一間隔でアドレスを飛ばして書き込みを行い、読み出しデータがブロックインタリーブされたものとなるような順序で読み出しを行って出力データとし、その後この読み出しと同じ順序で書き込みを行うことによりブロックインタリーブを行うようにしたので、1ブロックの記憶領域を持つ記憶装置1面でのブロックインタリーブ動作が可能となり、かつ記憶装置制御装置の回路規模が縮小可能となる効果がある。

【0137】

また、本願の請求項12の発明に係るブロックインタリーブ方法によれば、請求項11記載のブロックインタリーブ方法において、ブロック番号 b の0番目のアドレス $Ab(0)$ を0とし、ブロック番号 b の n 番目(n は0以上の整数)のアドレス $Ab(n)$ を、 α (α は、2以上の整数)と M の $(b-x)$ 乗(x は0以上 b 以下の整数)の乗算結果に $Ab(n-1)$ を加算した結果を $L \times M - 1$ で割った剰余から生成し、この生成したアドレスに対して $L \times M$ データの記憶装置の読み出し、書き込みを繰り返すことによりブロックインタリーブを行うようにしたので、記憶装置、アドレス生成装置を最適化でき、最小限の回路面積により、ブロックインタリーブを行うことが可能となる効果がある。

【0138】

また、本願の請求項13の発明に係るブロックインタリーブ方法によれば、請求項12記載のブロックインタリーブ方法において、前記 α と $L \times M - 1$ との間に公約数が存在しないように、これらの値を設定するようにしたので、アドレスの生成規則が破綻するのを防止でき、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックインタリーブを行うことが可能となる効果がある。

【0139】

また、本願の請求項14の発明に係るブロックインタリーブ方法によれば、請求項12記載のブロックインタリーブ方法において、前記 α と M の $(-x)$ 乗が等しくないように、これらの値を設定するようにしたので、最初の書き込みの時点においてアドレスを連続して書き込みを行うのを防止でき、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックインタリーブを行うことが可能となる効果がある。

【0140】

また、本願の請求項15の発明に係るブロックインタリーブ方法によれば、請求項12記載のブロックインタリーブ方法において、前記 α 、 L 、 M の値を、それぞれ $\alpha=20$ 、 $L=203$ 、 $M=8$ としたので、アドレス生成装置を構成する比較器の面積を低減でき、記憶装置、アドレス生成装置を最適化し、最小限の回路面積に

より、ブロックインタリーブを行うことが可能となる効果がある。

【0141】

また、本願の請求項16の発明に係るブロックデインタリーブ方法によれば、アドレスが連続する順序で割り当てられた記憶装置に対し1アドレス以上の同一間隔でアドレスを飛ばして書き込みを行い、読み出しデータがブロックデインタリーブされたものとなるような順序で読み出しを行って出力データとし、その後この読み出しと同じ順序で書き込みを行うことによりブロックデインタリーブを行うようにしたので、1ブロックの記憶領域を持つ記憶装置1面でのブロックデインタリーブ動作が可能となり、かつ記憶装置制御装置の回路規模が縮小可能となる効果がある。

【0142】

また、本願の請求項17の発明に係るブロックデインタリーブ方法によれば、請求項16記載のブロックデインタリーブ方法において、ブロック番号 b の0番目のアドレス $Ab(0)$ を0とし、ブロック番号 b の n 番目(n は0以上の整数)のアドレス $Ab(n)$ を、 α (α は、2以上の整数)と L の $(b-x)$ 乗(x は0以上 b 以下の整数)の乗算結果に $Ab(n-1)$ を加算した結果を $L \times M-1$ で割った剰余から生成し、この生成したアドレスに対して $L \times M$ データの記憶装置の読み出し、書き込みを繰り返すことによりブロックデインタリーブを行うようにしたので、記憶装置、アドレス生成装置を最適化でき、最小限の回路面積により、ブロックデインタリーブを行うことが可能となる効果がある。

【0143】

また、本願の請求項18の発明に係るブロックデインタリーブ方法によれば、請求項17記載のブロックデインタリーブ方法において、前記 α と $L \times M-1$ との間に公約数が存在しないように、これらの値を設定したので、アドレスの生成規則が破綻するのを防止でき、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックデインタリーブを行うことが可能となる効果がある。

【0144】

また、本願の請求項19の発明に係るブロックデインタリーブ方法によれば、請求項17記載のブロックデインタリーブ方法において、前記 α と L の $(-x)$ 乗が

等しくないように、これらの値を設定するようにしたので、最初の書き込みの時点においてアドレスを連続して書き込みを行うのを防止でき、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックデインタリーブを行うことが可能となる効果がある。

【0145】

また、本願の請求項20の発明に係るブロックデインタリーブ方法によれば、請求項17記載のブロックデインタリーブ方法において、前記 α 、 L 、 M の値を、それぞれ $\alpha=20$ 、 $L=203$ 、 $M=8$ としたので、アドレス生成装置を構成する比較器の面積を低減でき、記憶装置、アドレス生成装置を最適化し、最小限の回路面積により、ブロックデインタリーブを行うことが可能となる効果がある。

【図面の簡単な説明】

【図1】 本願発明の実施の形態1によるブロックインタリーブ装置の構成を示すブロック図。

【図2】 本願発明の実施の形態1によるブロックインタリーブ装置における記憶装置の書き込み読み出し順の一例を示す図。

【図3】 本願発明の実施の形態1によるブロックインタリーブ装置において記憶装置が1面で済む点を示すブロック図。

【図4】 従来および本願発明の実施の形態1によるブロックインタリーブ装置におけるオーバーフロー処理装置の構成を示す図。

【図5】 従来のブロックインタリーブ装置の記憶装置制御装置における比較器の構成を示す図。

【図6】 本願発明の実施の形態1によるブロックインタリーブ装置の記憶装置制御装置における比較器の構成を示す図。

【図7】 本願発明の実施の形態2によるブロックデインタリーブ装置の構成を示すブロック図。

【図8】 本願発明の実施の形態2によるブロックデインタリーブ装置における記憶装置の書き込み読み出し順の一例を示す図。

【図9】 本願発明の実施の形態2によるブロックデインタリーブ装置において記憶装置が1面で済む点を示すブロック図。

【図 10】 従来および本願発明の実施の形態 2 によるブロックデインタリーブ装置におけるオーバーフロー処理装置の構成を示す図。

【図 11】 従来のブロックデインタリーブ装置の記憶装置制御装置における比較器の構成を示す図。

【図 12】 本願発明の実施の形態 2 によるブロックデインタリーブ装置の記憶装置制御装置における比較器の構成を示す図。

【図 13】 従来のブロックインタリーブ装置およびブロックデインタリーブ装置における記憶装置の書き込み読み出し順を示す図。

【図 14】 従来のブロックインタリーブ装置およびブロックデインタリーブ装置において記憶装置が 1 面で済む点を示すブロック図。

【符号の説明】

4, 104 記憶装置

12, 112 制御装置

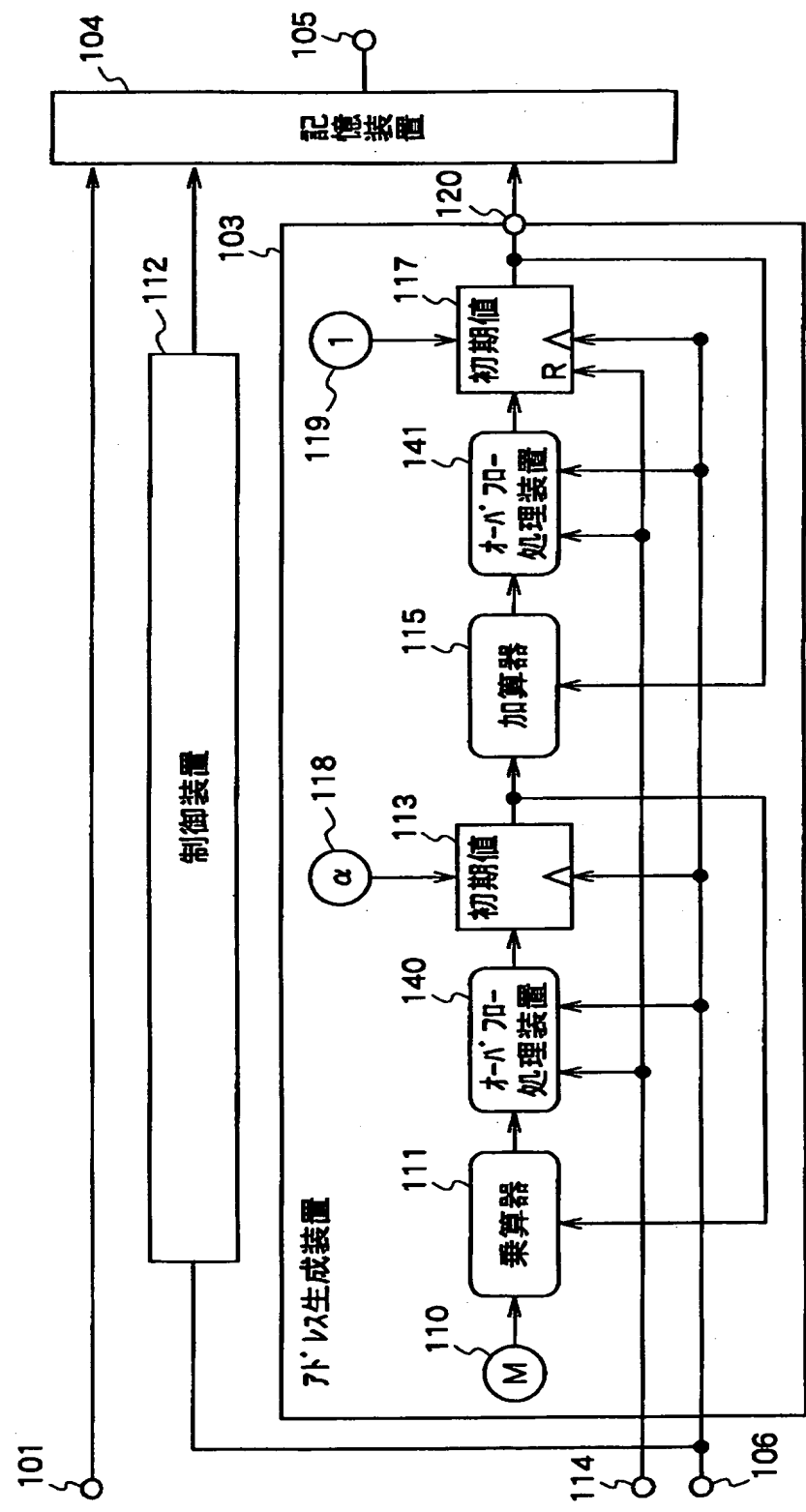
3, 103 アドレス生成装置

18, 19, 118, 119 定数発生器

112, 116, 140, 141 オーバフロー処理装置

【書類名】 図面

【図 1】

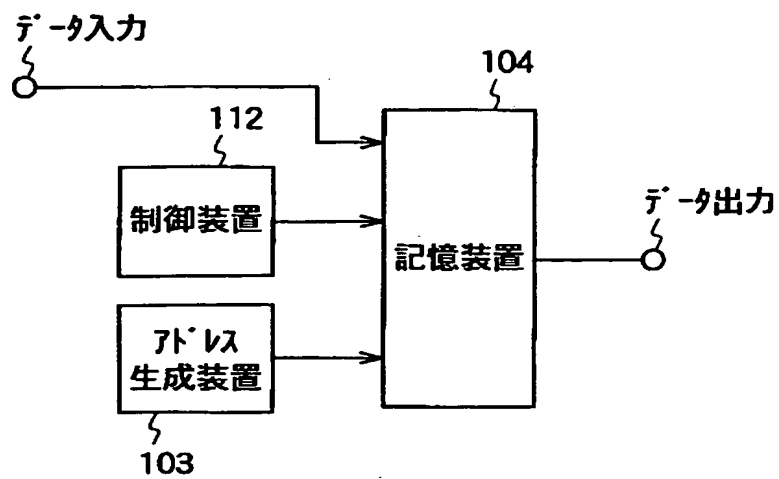


【図 2】

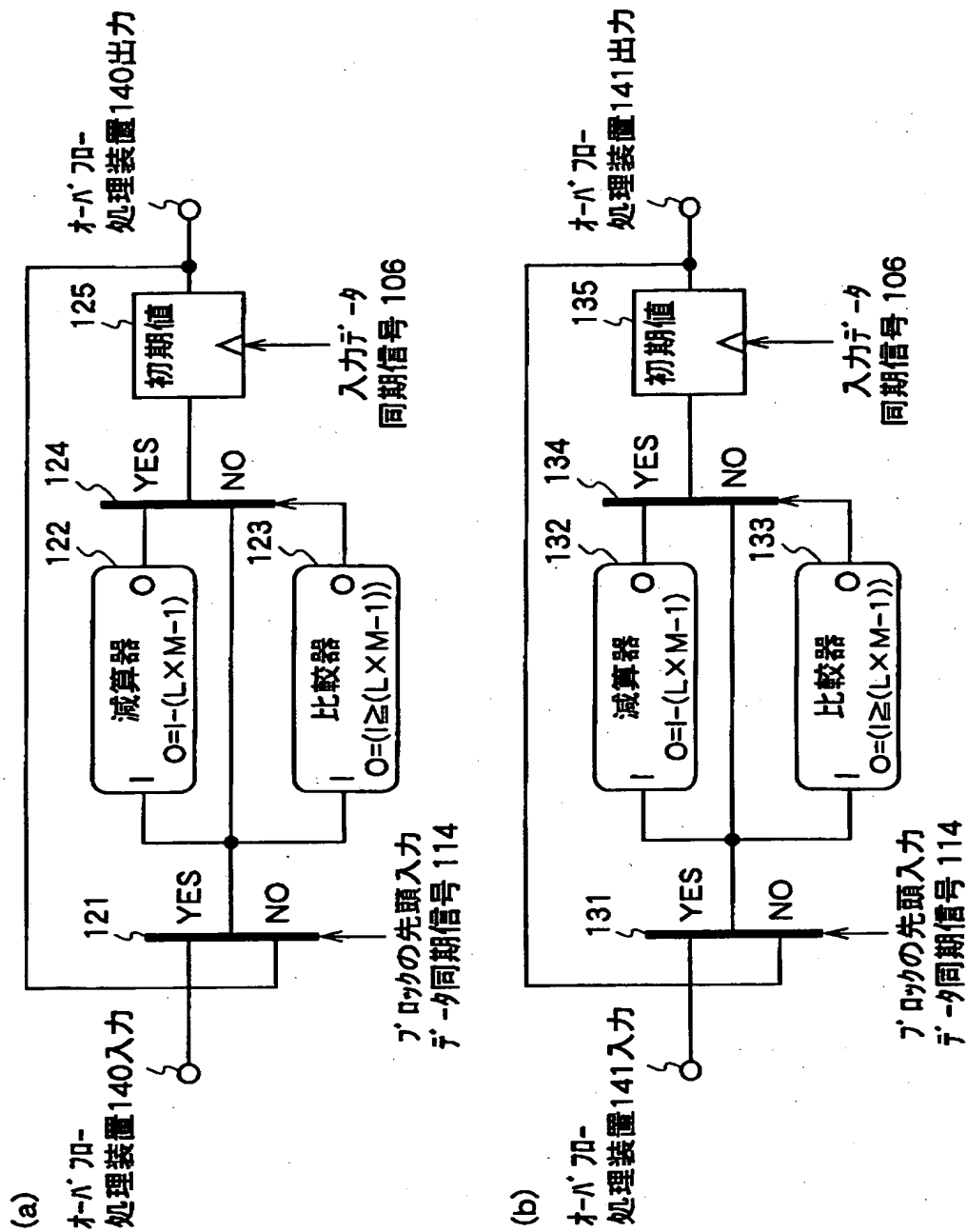
(a) 2	(b) 10	(c) $50-(19 \times 2)=12$	(d) $60-(19 \times 3)=3$	(e) 15
0 10 1 11 2	0 2 4 6 8	0 8 16 5 13	0 13 7 1 14	0 14 9 4 18
12 3 13 4 14	10 12 14 16 18	2 10 18 7 15	8 2 15 9 3	13 8 3 17 12
5 15 6 16 7	1 3 5 7 9	4 12 1 9 17	16 10 4 17 11	7 2 16 11 6
17 8 18 9 19	11 13 15 17 19	6 14 3 11 19	5 18 12 6 19	1 15 10 5 19

(f) $75-(19 \times 3)=18$	(g) $90-(19 \times 4)=14$	(h) $70-(19 \times 3)=13$	(i) $65-(19 \times 3)=8$	(j) $40-(19 \times 2)=2$
0 18 17 16 15	0 15 11 7 3	0 3 6 9 12	0 12 5 17 10	0 10 1 11 2
14 13 12 11 10	18 14 10 6 2	15 18 2 5 8	3 15 8 1 13	12 3 13 4 14
9 8 7 6 5	17 13 9 5 1	11 14 17 1 4	6 18 11 4 16	5 15 6 16 7
4 3 2 1 19	16 12 8 4 19	7 10 13 16 19	9 2 14 7 19	17 8 18 9 19

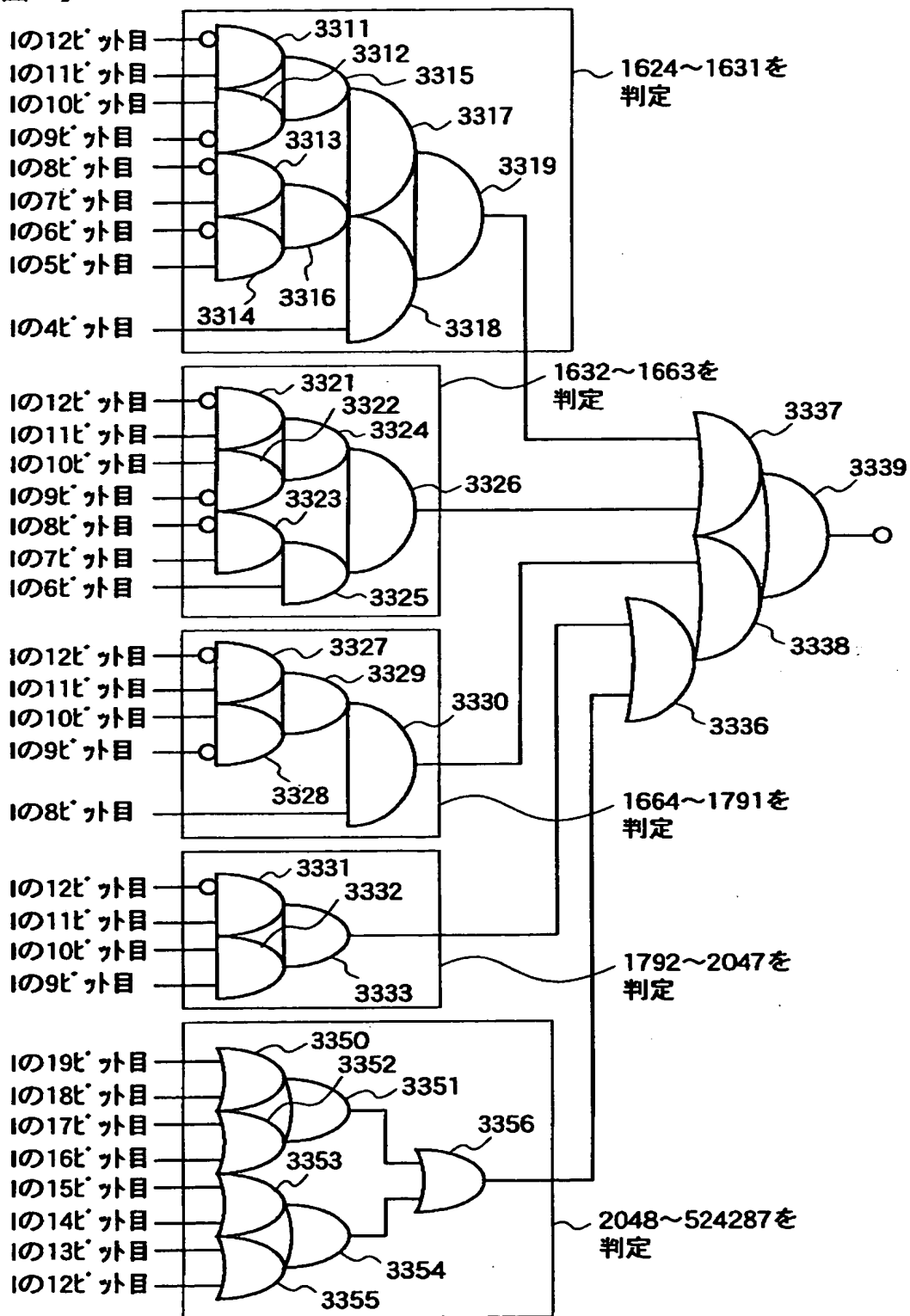
【図3】



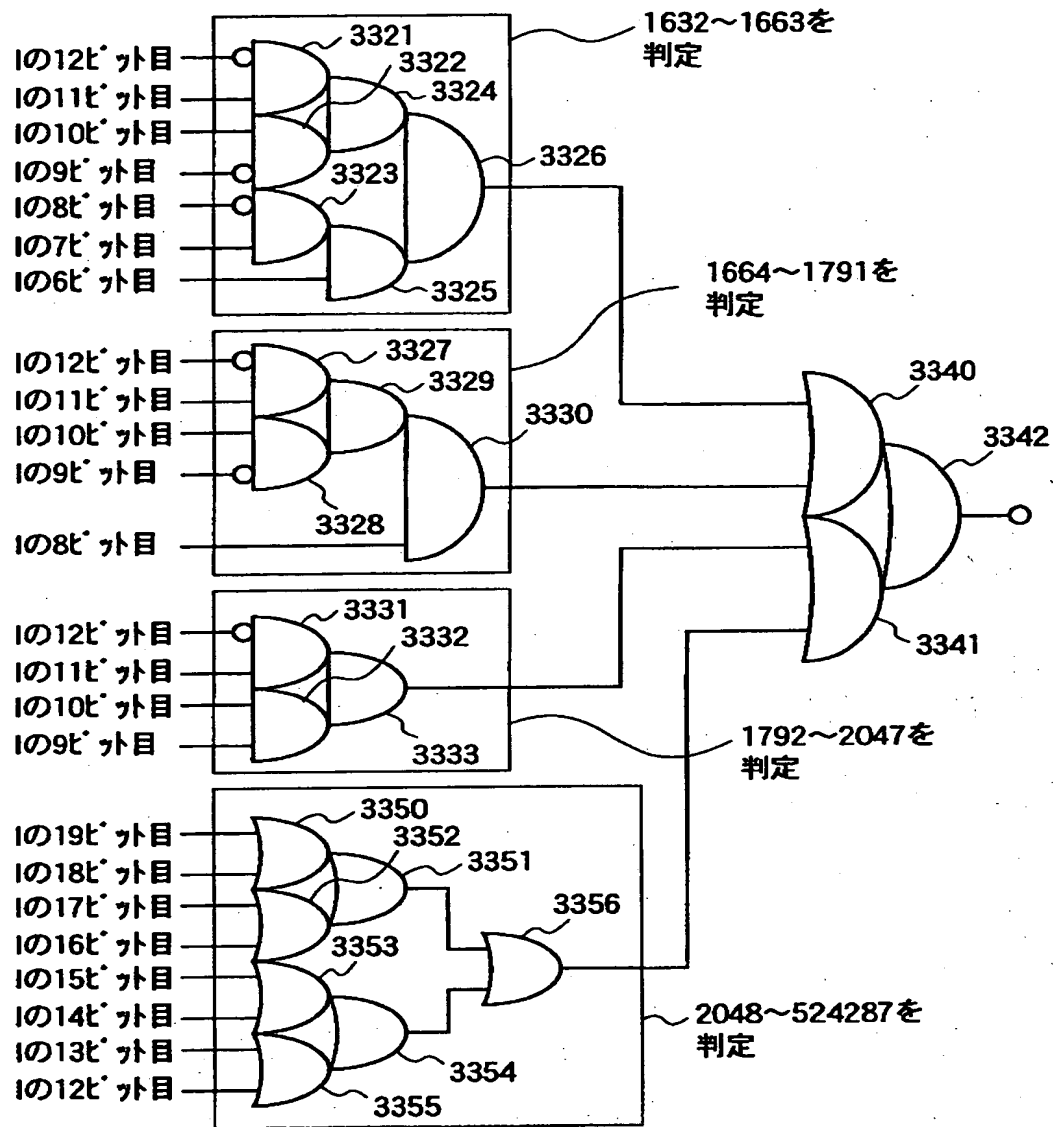
【図 4】



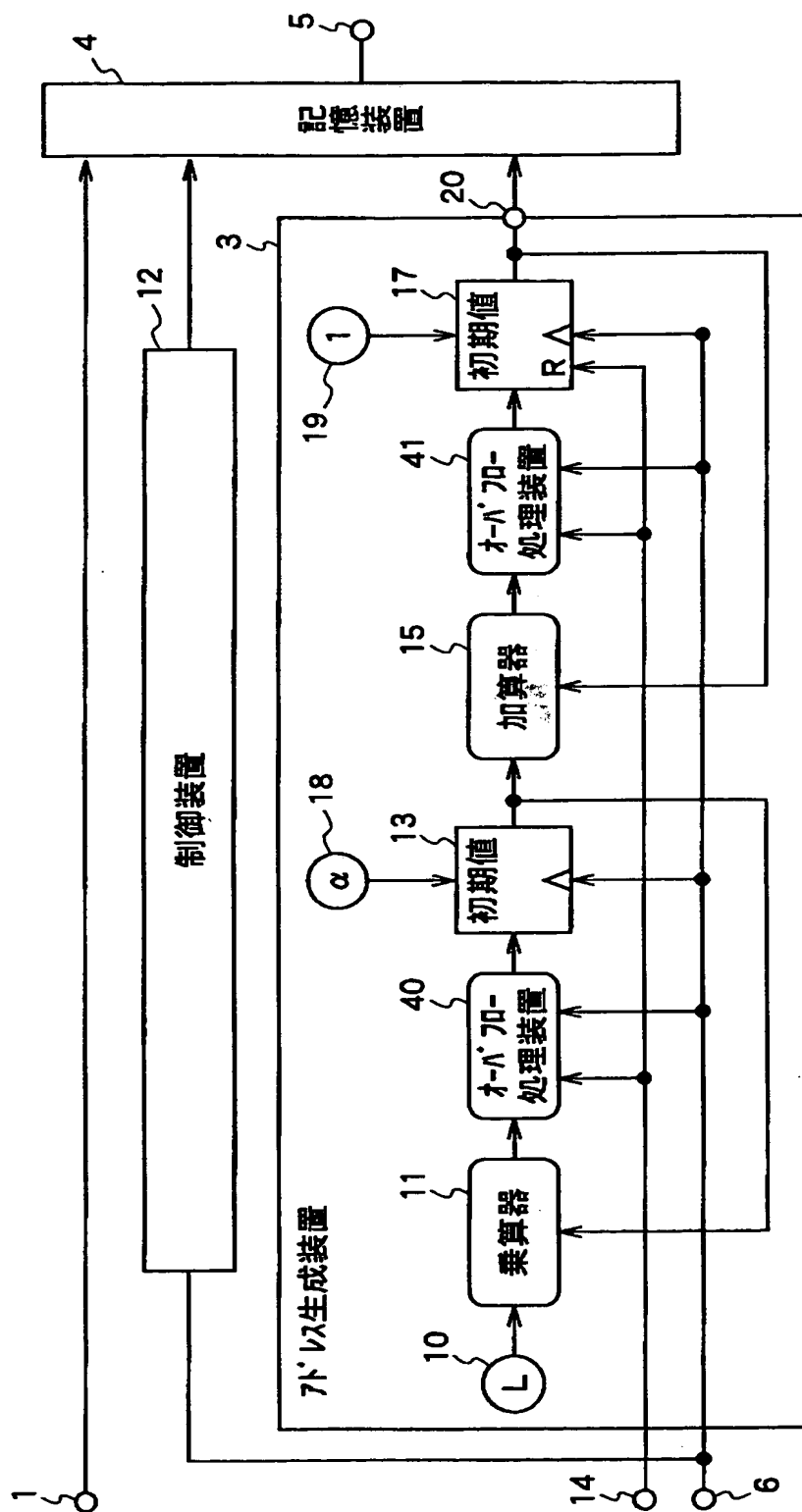
【図 5】



【図 6】



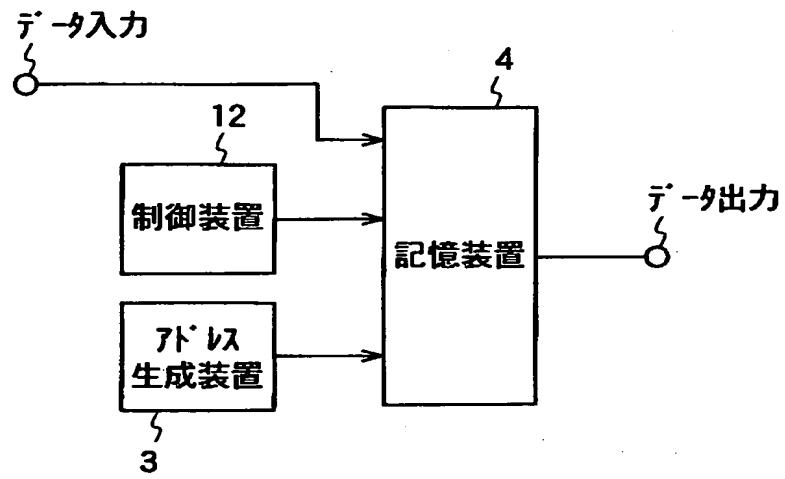
【図 7】



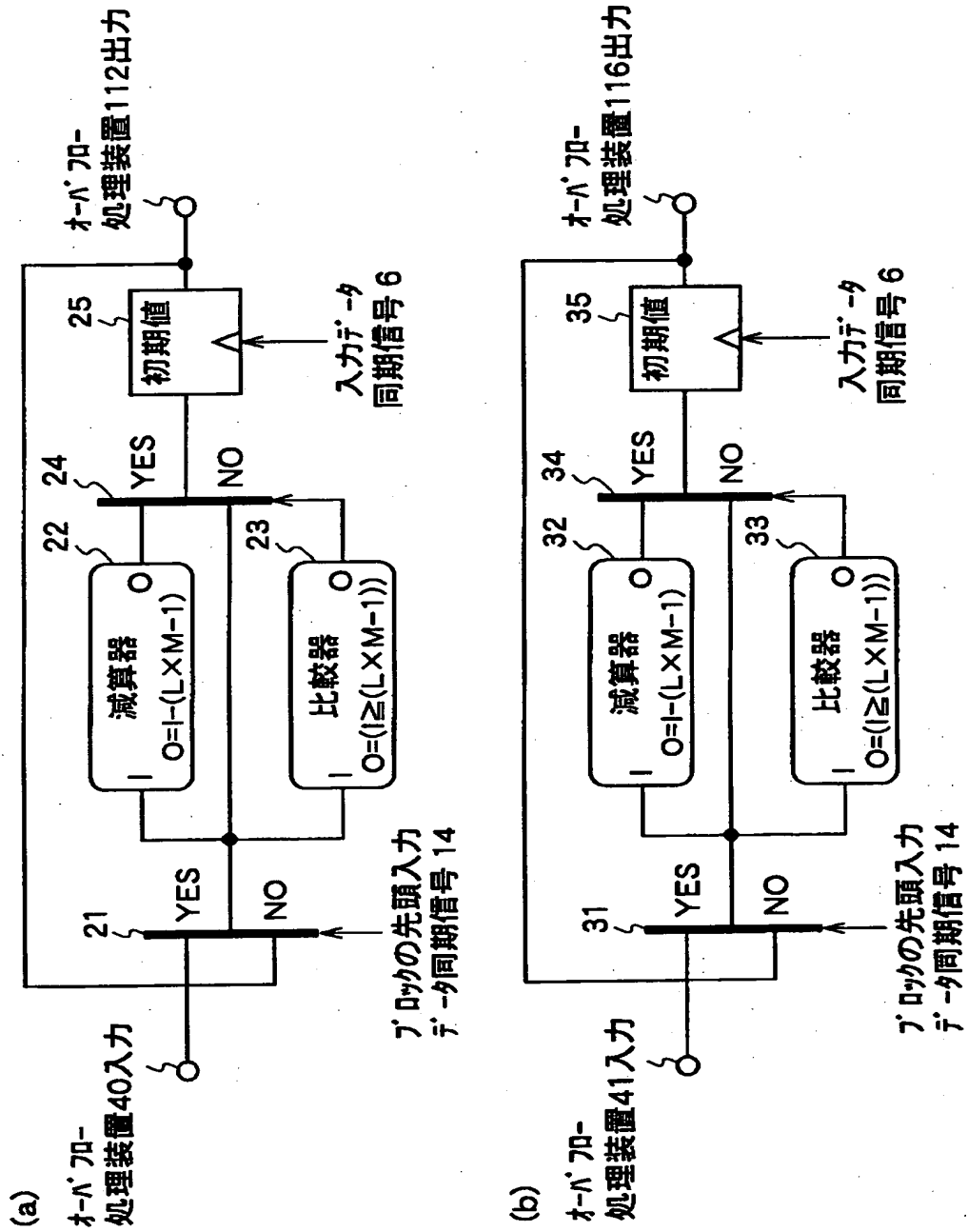
【図 8】

<p>(a) 2</p> <table border="1"> <tr><td>0</td><td>10</td><td>1</td><td>11</td><td>2</td></tr> <tr><td>12</td><td>3</td><td>13</td><td>4</td><td>14</td></tr> <tr><td>5</td><td>15</td><td>6</td><td>16</td><td>7</td></tr> <tr><td>17</td><td>8</td><td>18</td><td>9</td><td>19</td></tr> </table>	0	10	1	11	2	12	3	13	4	14	5	15	6	16	7	17	8	18	9	19	<p>(b) 8</p> <table border="1"> <tr><td>0</td><td>12</td><td>5</td><td>17</td><td>10</td></tr> <tr><td>3</td><td>15</td><td>8</td><td>1</td><td>13</td></tr> <tr><td>6</td><td>18</td><td>11</td><td>4</td><td>16</td></tr> <tr><td>9</td><td>2</td><td>14</td><td>7</td><td>19</td></tr> </table>	0	12	5	17	10	3	15	8	1	13	6	18	11	4	16	9	2	14	7	19	<p>(c) 32-19=13</p> <table border="1"> <tr><td>0</td><td>3</td><td>6</td><td>9</td><td>12</td></tr> <tr><td>15</td><td>18</td><td>2</td><td>5</td><td>8</td></tr> <tr><td>11</td><td>14</td><td>17</td><td>1</td><td>4</td></tr> <tr><td>7</td><td>10</td><td>13</td><td>16</td><td>19</td></tr> </table>	0	3	6	9	12	15	18	2	5	8	11	14	17	1	4	7	10	13	16	19	<p>(d) 52-(19×2)=14</p> <table border="1"> <tr><td>0</td><td>15</td><td>11</td><td>7</td><td>3</td></tr> <tr><td>18</td><td>14</td><td>10</td><td>6</td><td>2</td></tr> <tr><td>17</td><td>13</td><td>9</td><td>5</td><td>1</td></tr> <tr><td>16</td><td>12</td><td>8</td><td>4</td><td>19</td></tr> </table>	0	15	11	7	3	18	14	10	6	2	17	13	9	5	1	16	12	8	4	19	<p>(e) 56-(19×2)=18</p> <table border="1"> <tr><td>0</td><td>18</td><td>17</td><td>16</td><td>15</td></tr> <tr><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td></tr> <tr><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td></tr> <tr><td>4</td><td>3</td><td>2</td><td>1</td><td>19</td></tr> </table>	0	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	19
0	10	1	11	2																																																																																																				
12	3	13	4	14																																																																																																				
5	15	6	16	7																																																																																																				
17	8	18	9	19																																																																																																				
0	12	5	17	10																																																																																																				
3	15	8	1	13																																																																																																				
6	18	11	4	16																																																																																																				
9	2	14	7	19																																																																																																				
0	3	6	9	12																																																																																																				
15	18	2	5	8																																																																																																				
11	14	17	1	4																																																																																																				
7	10	13	16	19																																																																																																				
0	15	11	7	3																																																																																																				
18	14	10	6	2																																																																																																				
17	13	9	5	1																																																																																																				
16	12	8	4	19																																																																																																				
0	18	17	16	15																																																																																																				
14	13	12	11	10																																																																																																				
9	8	7	6	5																																																																																																				
4	3	2	1	19																																																																																																				
<p>(f) 72-(19×3)=15</p> <table border="1"> <tr><td>0</td><td>14</td><td>9</td><td>4</td><td>18</td></tr> <tr><td>13</td><td>8</td><td>3</td><td>17</td><td>12</td></tr> <tr><td>7</td><td>2</td><td>16</td><td>11</td><td>6</td></tr> <tr><td>1</td><td>15</td><td>10</td><td>5</td><td>19</td></tr> </table>	0	14	9	4	18	13	8	3	17	12	7	2	16	11	6	1	15	10	5	19	<p>(g) 60-(19×3)=3</p> <table border="1"> <tr><td>0</td><td>13</td><td>7</td><td>1</td><td>14</td></tr> <tr><td>8</td><td>2</td><td>15</td><td>9</td><td>3</td></tr> <tr><td>16</td><td>10</td><td>4</td><td>17</td><td>11</td></tr> <tr><td>5</td><td>18</td><td>12</td><td>6</td><td>19</td></tr> </table>	0	13	7	1	14	8	2	15	9	3	16	10	4	17	11	5	18	12	6	19	<p>(h) 12</p> <table border="1"> <tr><td>0</td><td>8</td><td>16</td><td>5</td><td>13</td></tr> <tr><td>2</td><td>10</td><td>18</td><td>7</td><td>15</td></tr> <tr><td>4</td><td>12</td><td>1</td><td>9</td><td>17</td></tr> <tr><td>6</td><td>14</td><td>3</td><td>11</td><td>19</td></tr> </table>	0	8	16	5	13	2	10	18	7	15	4	12	1	9	17	6	14	3	11	19	<p>(i) 48-(19×2)=10</p> <table border="1"> <tr><td>0</td><td>2</td><td>4</td><td>6</td><td>8</td></tr> <tr><td>10</td><td>12</td><td>14</td><td>16</td><td>18</td></tr> <tr><td>1</td><td>3</td><td>5</td><td>7</td><td>9</td></tr> <tr><td>11</td><td>13</td><td>15</td><td>17</td><td>19</td></tr> </table>	0	2	4	6	8	10	12	14	16	18	1	3	5	7	9	11	13	15	17	19	<p>(j) 40-(19×2)=2</p> <table border="1"> <tr><td>0</td><td>10</td><td>1</td><td>11</td><td>2</td></tr> <tr><td>12</td><td>3</td><td>13</td><td>4</td><td>14</td></tr> <tr><td>5</td><td>15</td><td>6</td><td>16</td><td>7</td></tr> <tr><td>17</td><td>8</td><td>18</td><td>9</td><td>19</td></tr> </table>	0	10	1	11	2	12	3	13	4	14	5	15	6	16	7	17	8	18	9	19
0	14	9	4	18																																																																																																				
13	8	3	17	12																																																																																																				
7	2	16	11	6																																																																																																				
1	15	10	5	19																																																																																																				
0	13	7	1	14																																																																																																				
8	2	15	9	3																																																																																																				
16	10	4	17	11																																																																																																				
5	18	12	6	19																																																																																																				
0	8	16	5	13																																																																																																				
2	10	18	7	15																																																																																																				
4	12	1	9	17																																																																																																				
6	14	3	11	19																																																																																																				
0	2	4	6	8																																																																																																				
10	12	14	16	18																																																																																																				
1	3	5	7	9																																																																																																				
11	13	15	17	19																																																																																																				
0	10	1	11	2																																																																																																				
12	3	13	4	14																																																																																																				
5	15	6	16	7																																																																																																				
17	8	18	9	19																																																																																																				

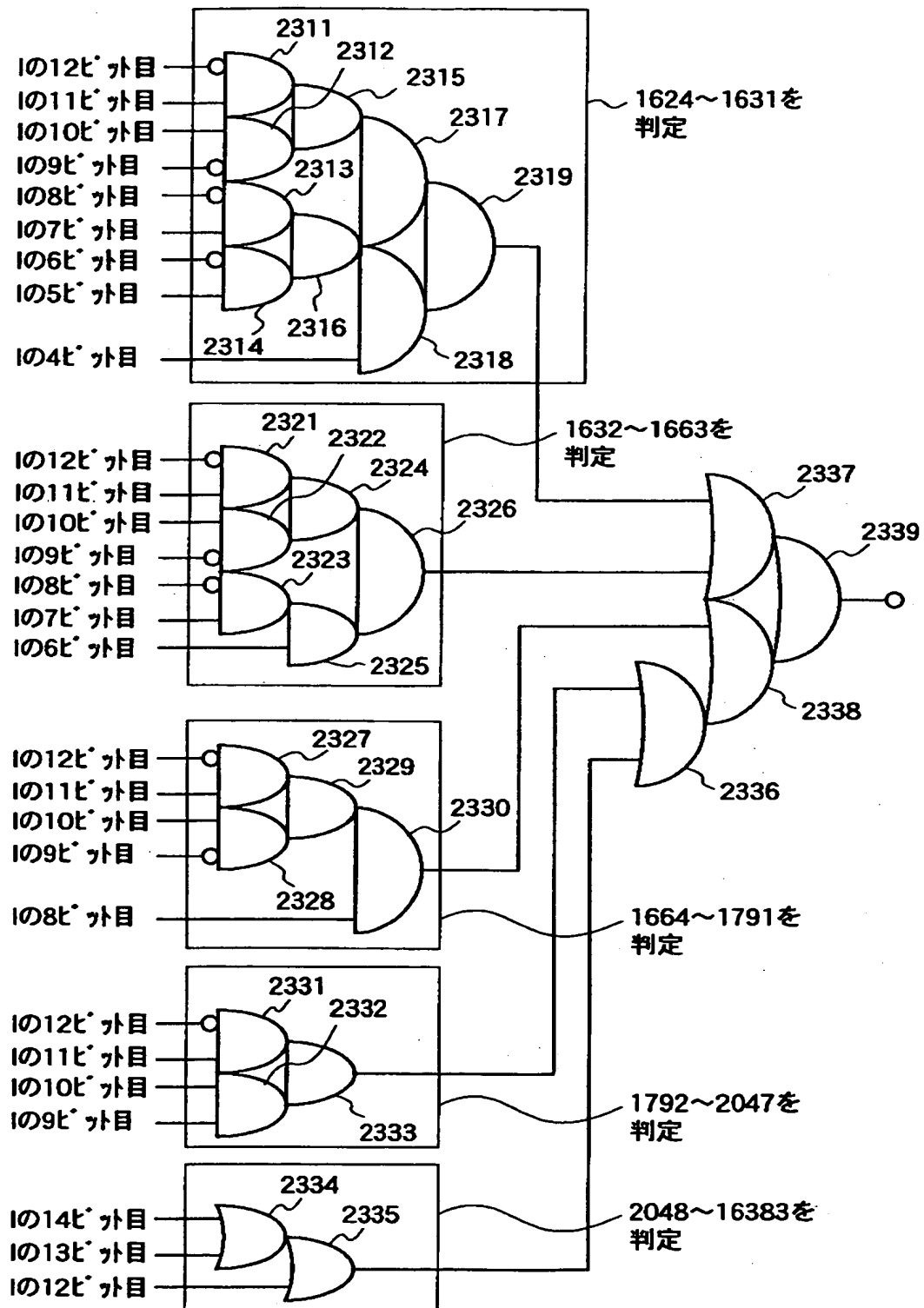
【図 9】



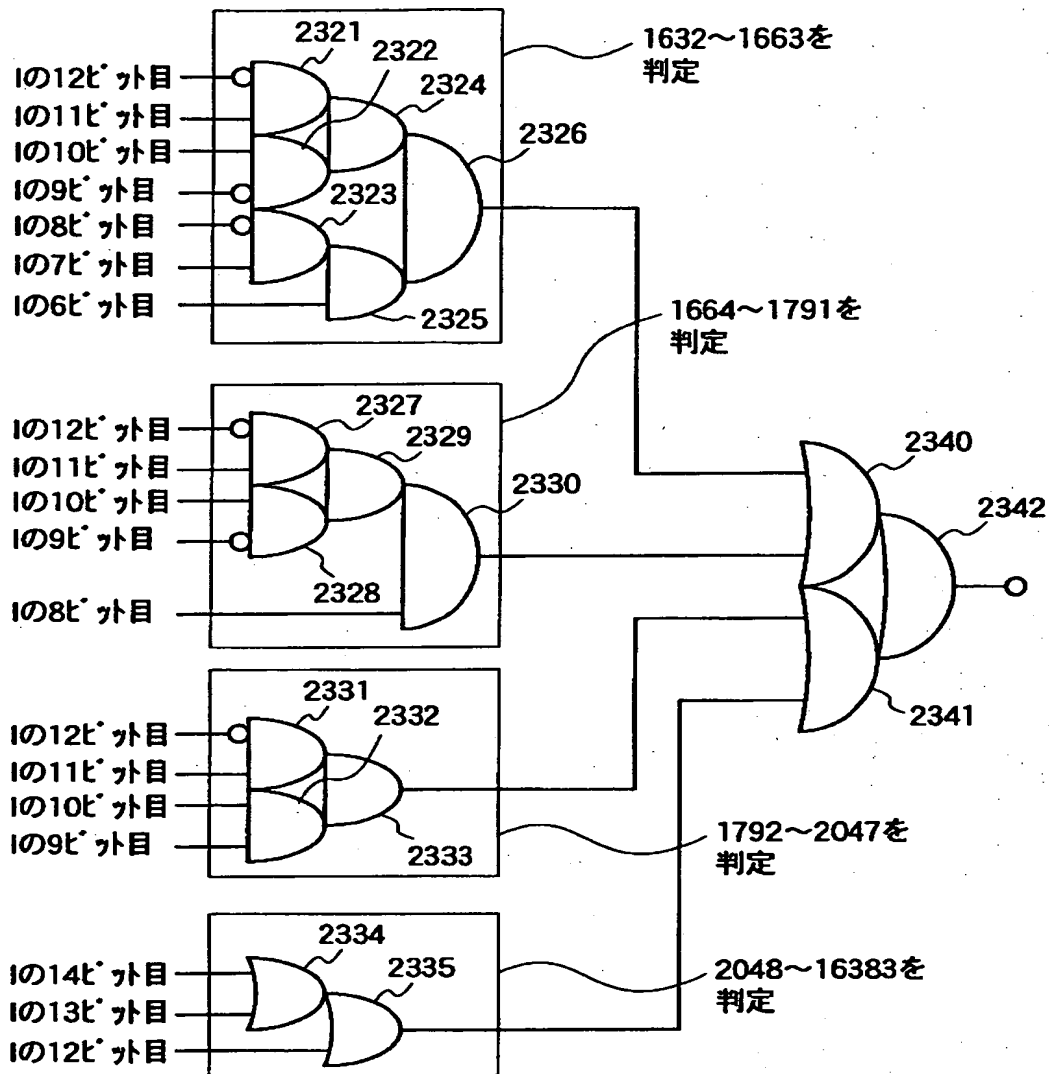
【図 10】



【図 1 1】



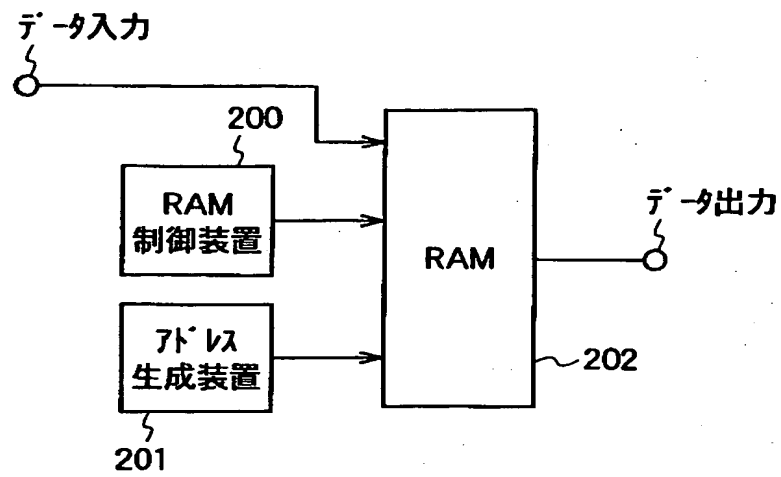
【図 1 2】



【図 13】

(a) 1	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19	(b) 5	0 4 8 12 16 1 5 9 13 17 2 6 10 14 18 3 7 11 15 19	(c) 25-19=6	0 16 13 10 7 4 1 17 14 11 8 5 2 18 15 12 9 6 3 19	(d) 30-19=11	0 7 14 2 9 16 4 11 18 6 13 1 8 15 3 10 17 5 12 19	(e) 55-(19×2)=17	0 9 18 8 17 7 16 6 15 5 14 4 13 3 12 2 11 1 10 19
(f) 85-(19×4)=9	0 17 15 13 11 9 7 5 3 1 18 16 14 12 10 8 6 4 2 19	(g) 45-(19×2)=7	0 11 3 14 6 17 9 1 12 4 15 7 18 10 2 13 5 16 8 19	(h) 35-19=16	0 6 12 18 5 11 17 4 10 16 3 9 15 2 8 14 1 7 13 19	(i) 80-(19×4)=4	0 5 10 15 1 6 11 16 2 7 12 17 3 8 13 18 4 9 14 19	(j) 20-19=1	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
(k) 1	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19	(l) 4	0 5 10 15 1 6 11 16 2 7 12 17 3 8 13 18 4 9 14 19	(m) 16	0 6 12 18 5 11 17 4 10 16 3 9 15 2 8 14 1 7 13 19	(n) 64-(19×3)=7	0 11 3 14 6 17 9 1 12 4 15 7 18 10 2 13 5 16 8 19	(o) 28-19=9	0 17 15 13 11 9 7 5 3 1 18 16 14 12 10 8 6 4 2 19
(p) 36-19=17	0 9 18 8 17 7 16 6 15 5 14 4 13 3 12 2 11 1 10 19	(q) 68-(19×3)=11	0 7 14 2 9 16 4 11 18 6 13 1 8 15 3 10 17 5 12 19	(r) 44-(19×2)=6	0 16 13 10 7 4 1 17 14 11 8 5 2 18 15 12 9 6 3 19	(s) 24-19=5	0 4 8 12 16 1 5 9 13 17 2 6 10 14 18 3 7 11 15 19	(t) 20-19=1	0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

【図 1 4】



【書類名】 要約書

【要約】

【課題】 1ブロックの記憶領域を持つ記憶装置1面でブロックインタリーブ、ブロックデインタリーブを行うブロックインタリーブ装置、ブロックデインタリーブ装置、ブロックインタリーブ方法およびブロックデインタリーブ方法において、より回路面積の削減および低消費電力化を実現すること。

【解決手段】 記憶装置104を制御する制御装置112およびアドレス生成装置103により、最初に書き込みを行う際に1つ以上アドレスを飛ばして書き込みを行うように、構成した。

【選択図】 図1

出 願 人 履 歴 情 報

識別番号

[000005821]

1. 変更年月日 1990年 8月28日

[変更理由] 新規登録

住 所 大阪府門真市大字門真1006番地

氏 名 松下電器産業株式会社

THIS PAGE BLANK (USPTO)